Chris Proctor

# Teaching statement

Even though I am no longer a full-time K-12 teacher, that work continues to shape my research and my professional identity. The context and methods of my research depend on sustained relationships with students, teachers, schools, and their communities. The theoretical ideas I find powerful are those which articulate what I see in classrooms: ideas alive in communities of practice; computers as tools and partners in thought; the mutual emergence of identity and participation; literacy spaces where students can be creators and audience. And my teaching practice at the undergraduate and graduate level is heavily informed by my years as a middle- and high-school teacher.

There is an intrinsic inside/outside duality to teaching. The teacher must on one hand be a designer: setting goals, developing curricula, practices, and tools to meet those goals, and constantly assessing how well they are working. But the teacher must also be present as a learner, re-discovering ideas alongside students and struggling anew with challenging ideas. I mean this! Even in a large lecture course, a professor will not be effective if he cannot get himself into the position of the learner. Powerful teaching and learning happen within human relationships and communities, and if a teacher is going to participate in helping students learn, he needs to be present to them as a co-learner. This demands openness and vulnerability.

A teacher's work as a designer starts with setting goals—not just what students are going to learn, but also what is meant by learning. This is always grounded in implicit or explicit theories of learning. In our paper[1] at this year's International Computing Education Research (ICER) conference, we argued that computer science education ought to support dialogue between cognitive, situated, and critical theories of learning. In my own teaching, I find literacy to be a useful construct for putting these framings in dialogue. Certainly, I want my students to leave class with new skills and knowledge. These need to be clearly visible and students need opportunities to practice and integrate them into their prior knowledge. At the same time, I need to cultivate communities of practice where students can build identities and participate in the real work of the field. And I want to connect the course to the real lives and futures of my students. Critical questions about who is benefitting and in what ways need to be part of the conversation.

Historically, university-level teaching has gotten away without thinking too hard about theories of learning. Success for students meant a good grade in the class; success for professors meant good course reviews. Coming from a K-12 teaching context, where teachers are charged with ensuring every student's success, the limitations of this approach are readily apparent. In my future university-level teaching, I am eager to take on pedagogical design challenges such as:

- Studio- or lab-based courses in design-based research, especially in which computational tools are developed, deployed in real-world settings, and used as research instruments.

1 Kafai, Y., Proctor, C., & Lui, D. (2019). From Theory Bias to Theory Dialogue: Embracing Cognitive, Situated, and Critical Framings of Computational Thinking in K-12 CS Education. In McCartney, R., Petersen, A., Robins, A., & Moskal, A. (Eds.) *Proceedings of the 2019 ACM Conference on International Computing Education Research.* (pp. 101-109) New York: ACM. **Received best paper award.**

- Introductory computational methods courses for graduate students in Education, including data science, natural language processing, quantitative ethnography, and learning analytics.
- The history and future of educational technology and infrastructure, from a learning sciences and/or science, technology, and society (STS) perspective. I am interested in considering the theoretical foundations of designs, their phenomenological and epistemological consequences, and how we might prepare for future developments such as automation and artificial intelligence.
- English or computer science teacher preparation.

However, a teacher is not a clockmaker who designs mechanisms and then leaves them to run. Building knowledge and practices together requires that my students and I humanize each other. I need to know where my students are coming from, how they regard the topics in the course, what they hope to get out of it. They need to see me as someone who also makes mistakes, who struggles to find unity in new ideas, as someone with his own history and hopes for the future. We need to build, together, an understanding of what it is to do computer science.

This is complicated by power, privilege, and positionality. What if I am the reason a student does not feel comfortable participating in class? In my teaching and in my mentorship of graduate students, I commit to doing my best to create an environment of trust, to make space for listening, where we can surface inequities and oppression and do our best to respond to them. This means decentering my approach to the curriculum, decentering my role in the classroom, explicitly questioning the hierarchies of status that order our interactions: race, gender, social class, educational credentials. It means taking active steps to make marginalized students feel welcome, and allowing myself to be open and vulnerable as well. I am queer. When and how do I share this with my students?

How to reconcile these two sides of teaching? My teacher preparation program left me with the phrase "be warmly demanding," suggesting a teacher should be both inside and outside. The two sides also merge through reflective practice and formative assessment. We tend to use stories about ourselves as learners as a model for pedagogy; a collaborative, reflective practice can help us become more aware of these implicit models and how they bias our pedagogical choices. Similarly, collecting lots of data about our students (and analyzing it with them) can provide dispassionate feedback about what is working for our students. In all the contexts where I have taught, excellence emerged through constant collaborative, reflective iteration.

My teaching has received recognition. In addition to teaching awards and top ratings on course evaluations, I have been honored by former students telling me it made a difference to them. A former middle-school computer science student told me that the flexibility with which we used computational ideas to represent ideas later helped them understand their sexuality in high school. Another student, who unexpectedly emailed me years after I taught him English in high school, wrote:

> Not sure if you remember me, but I was a student in your class at X. Just wanted to reach out and tell you that you were my favorite teacher at X and at the time you made a huge difference in my life. I hated school (combination of not fitting in, constant stress, and lack

> of motivation) but thoroughly enough enjoyed your class and your teaching style, and always tried to make it on time and do my work.

In student feedback on "Beyond Bits and Atoms: Designing Digital Tools," a graduate course I co-taught with my advisor Paulo Blikstein[2], a student wrote:

> Oh man. I had always been scared of lab tools / thought I was someone who just sucked at building physical things. This class is scaffolded so well, and reflecting back I'm blown away at how much more familiar I feel with all these tools, and how much more confident I feel as a maker. Chris was perhaps the nicest, most supportive teacher I've ever had.

My teaching practice, and the research that has grown out of it, are central to my identity. My core ethical principle is to believe in the potential of others to grow and change, that we always only see a limited slice of a person, and that given the right conditions, anyone can grow and thrive. I view teaching as a means of using my power toward these ends.

2 Average student rating: 4.9/5 (department average: 4.2/5)