Teaching Statement
Andrew Head

December 13, 2020

Within the universe of outcomes of computer science pedagogy, one goal that has always resonated with me is helping students learn how to design and engineer impactful technical systems. My teaching is structured around a belief that instruction in STEM education is often made effective through objective-based directed practice and targeted feedback, grounded in real-world interests and applications.

In this statement, I describe how I actively apply this philosophy as a course instructor, research mentor, and tools researcher. Then I conclude with a discussion of courses I am prepared to teach and my preparedness to conduct remote instruction.

Lecturing

My most formative teaching experience was when I co-lectured *Computer Science 160*, a 76-student upper division course on user interface design and implementation. During the course, I refined my approach to teaching, ultimately receiving an **Outstanding Graduate Student Instructor Award, and teaching effectiveness ratings matching or surpassing those in recent offerings of the course**.

Figure 1: I design assignments that are grounded in real-world applications and interests. Students in my class have designed interactive systems that help (A) writers manage versions of their prose, (B) joggers plan out jogging routes, (C) athletic coaches sketch out plays, (D) older adults with Alzheimer's preserve their memories.

My approach to teaching can be best summarized by the application of five research-based principles. I leverage these same principles whether I am teaching students how to write programs, solve mathematical problems, or conduct usability studies:

Objectives: I use learning objectives to plan instructional content and motivate students. For example, I start planning lectures by determining specific capabilities students should acquire (e.g., "students will be able to design visualizations that can be readily used by others in order to answer questions"). Recognizing that priming with objectives can lead to better learning,[1] I express these objectives in lecture, and integrate them into the syllabus and project descriptions.

[1] Hamaker. "The effects of adjunct questions on prose learning". In: *Review of educational research* 56.2 (1986)

Grounding in real-world interests and applications: One source of motivation for students can come from a student's expectation that they'll be able apply what they've learned in class to their own goals.[2] I design projects to help students understand how the concepts learned in class relate to their interests. For instance, in Computer Science 160, students design an interactive application for a user group of their choice (Figure 1). By interviewing and evaluating their ideas with users, students have a chance to see how deep user research and

[2] Wigfield and Eccles. "The development of achievement task values: A theoretical analysis". In: *Developmental review* 12.3 (1992)

careful prototyping lead to systems that excite users.

Directed practice: I design classroom activities and projects to give students directed practice of component skills in isolation.[3] Before asking students to observe users outside the classroom, I have them practice observations of each other using interfaces through active learning activities in the classroom. I have them do the same with other methods like interviewing and running controlled experiments (Figure 2). Before undertaking open-ended projects, students complete short "implementation prototypes"[4] to learn user interface programming concepts and toolkits they will use in their projects.

Figure 2: Students practicing controlled user studies of user interfaces in the Computer Science 160 classroom.

[3] Salden, Paas, and Merriënboer. "A comparison of approaches to learning task selection in the training of complex cognitive skills". In: *Computers in Human Behavior* 22.3 (2006)

[4] Houde and Hill. "What do prototypes prototype?" In: *Handbook of human-computer interaction*. Elsevier, 1997

Targeted (and scalable) feedback: Recognizing that students benefit from feedback that is targeted,[5] timely,[6] and frequent, I design curricula to provide a high quantity of peer feedback and high quality instructor feedback. For instance, to help students improve their plans for studies with user groups, I designed a day's lecture that included a ten-minute mini-lecture providing generalized feedback, followed by ten minutes of dedicated coaching between each team and one member of the teaching staff, and having students spend the remainder of the time role-playing their study plans with their peers.

[5] Balzer, Doherty, et al. "Effects of cognitive feedback on performance." In: *Psychological bulletin* 106.3 (1989)

[6] Hattie and Timperley. "The power of feedback". In: *Review of educational research* 77.1 (2007)

Growth mindset: Many students taking my classes begin the class thinking that they are "bad" at programming or design. Knowing that "malleable mindsets" can be taught and that they lead to better learning outcomes,[7] I encourage students to think about their skills as process rather than raw ability. For open-ended projects, rubrics are designed to reward growth by assigning points based on thoughtfulness and revision, rather than technical complexity or visual flair. I convey my belief that anyone can design to my students by showing a sense of humor about my own messy but entirely functional sketching ability on the first day of class.

[7] Aronson, Fried, and Good. "Reducing the effects of stereotype threat on African American college students by shaping theories of intelligence". In: *Journal of experimental social psychology* 38.2 (2002)

Research mentoring

As a research mentor, I bring these same principles to bear in guiding students to conduct leading research. **I have honed my ability to mentor students by advising dozens of students across the doctoral, Master's, and Bachelor's levels**.

My experiences mentoring Master's thesis student Kunal Chaudhary and Bachelor's student and UC LEADS scholar Jason Jiang show how I provide steady support for young researchers throughout the research life cycle. Both students developed innovative interactive systems of their own design from a blank slate as part of independent research. Kunal design and evaluated a tool for data scientists to browse an overabundance of results produced from data analysis code in computational notebooks. Jason designed and evaluated a tool to help data scientists keep their computational notebooks clean through lightweight archiving of code (Figure 3). For both students, my role was to help them identify promising research ideas and guide them in acquiring and applying disciplinary skills such as literature review, rapid prototyping, and evaluating interactive systems.

At the doctoral level, I have mentored Nate Weinman from UC Berkeley, and Raymond Fok, a student advised by Daniel S. Weld and James Fogarty at the University of Washington. I helped Weinman and Fok develop disciplinary skills in human-computer interaction such as selecting research ideas, qualitative analysis, and study design, through frequent meetings and feedback on research. Fok became third author on our recent manuscript for the ScholarPhi tool, and is now leading follow-up research on the topic.

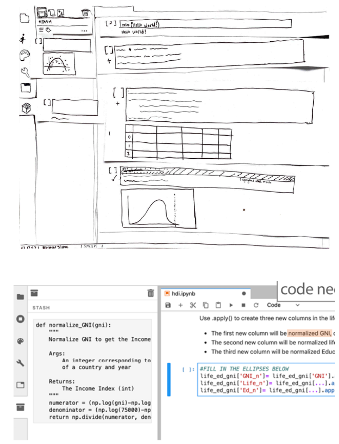

Figure 3: A tool developed by my mentee Jason Jiang, from its inception (paper prototype, top), to its implementation and evaluation (bottom).

Tools research

My research is motivated by a desire to grant learners access to high-quality learning experiences in massive classrooms, and outside the classroom. I therefore have developed tools to scale feedback and engage experts in interaction with students (see Section 3 of my research statement) and to help teachers author instructional content such as programming tutorials more efficiently (Section 1).

Courses

At the undergraduate level, I am prepared to teach courses on **User Interface Design and Implementation**, **Information Visualization**, and **Software Engineering**. Given time to prepare, I can also teach courses on **Data Science** and **Program Analysis**.

At the graduate level, I can envision teaching **Research Methods in Human-Computer Interaction** and **Empirical Software Engineering**. I particularly look forward to developing curriculum at the intersection of human-computer interaction, data science, software tools, and AI, including courses on **Designing Usable Data Science Tools**; **Intelligent Tools for Education**; and **Human-AI Creativity Tools**.

Remote instruction

In the time of shelter-in-place, I have educated myself about and gained experience conducting peer learning activities over Zoom. For courses with a design component, I plan to teach students how to recruit, conduct, and learn from user research remotely based on lessons I have learned and shared with my mentees in my own research.[8]

[8] Head, Lo, Kang, Fok, Skjonsberg, Weld, and Hearst. "Augmenting Scientific Papers with Just-in-Time, Position-Sensitive Definitions of Terms and Symbols". In: *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 2021. arXiv: 2009.14237 [cs.HC]. (Demo video). To appear.