Teaching Statement

Michael Correll

Background

As a multi-disciplinary teacher, my belief is that structuring a student's ability to *think* is the core of Science, Technology, Engineering, and Mathematics (STEM) education. With proper understanding of logical thinking, abstraction, and encapsulation, learning new concepts and perspectives is easy; without it, we risk creating students who may be very skilled at one particular task, but cannot adapt to the changing needs of the job market or their own intellectual strivings. As an undergraduate, I completed dual majors in Philosophy and Computer Science, and also completed a minor in Mathematics. Each of these disciplines focuses on logical thinking from a different perspective, but shares a commonality of approaches and pedagogical needs. Over the years, I applied these perspectives in my teaching experiences in both introductory and graduate level courses across many departments, and diverse classrooms.

Teaching Experience

My first teaching experience was as a teaching assistant in Johns Hopkins' Center for Talented Youth summer program for gifted and talented middle- and high-schoolers. As part of the Introduction to Robotics course (and later, the Mathematical Reasoning course), I was able to work with promising students at the very beginning of their STEM education, building up concepts like iteration, recursion, and induction that lay at the heart of logical, programmatic, and mathematical thinking. Over the course of a summer, middle school students (the vast majority of whom had no prior formal training in programming) were able to solve both engineering and programming challenges, including competing in a robotic sumo wrestling competition, and participating in a robot obstacle course.

As an undergraduate, I was a grader and class assistant for courses in three departments. I assisted with introductory logic and critical thinking in Philosophy, statistics in Mathematics, and with the introductory course in Computer Science. I gained experience with grading, assisting in lectures, and running educational computer labs. I take pride in being able to witness, in all of these settings, "Eureka" moments from students who had never before been exposed to symbolic logic, probability, and programming, but who then had mastery over the foundational tools they would use in the rest of their careers.

In graduate school, I was a teaching assistant for an introductory calculus course, where I was able to design my own lesson plans in concert with the professor, reinforcing the material initially presented in a 200-person lecture in smaller, collaborative classrooms. I was also a teaching assistant for a senior level computer game technologies course. The capstone of this course was a multi-week game jam, a new and valuable experience for the students, many of whom had no prior experience with collaborative software engineering. Among other projects, the students were able to create a 3D racing game, an RPG based on musical notation, and a physics-based curling simulator.

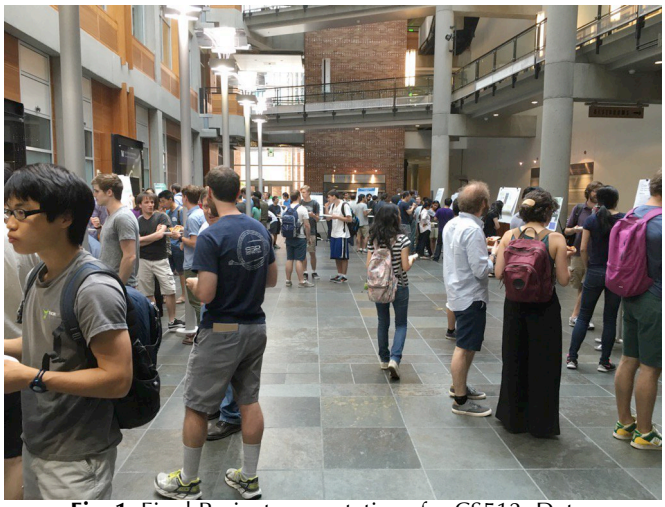

Fig .1: Final Project presentations for CS512: Data Visualization at the University of Washington. Projects ranged from an interactive photo browser designed to highlight the effects of climate change, to a 3D navigation tool for summarizing brain imaging data.

My most recent teaching experiences was as teaching assistant for the graduate level information visualization course at Madison, and as part of the instructional staff for the equivalent course at Washington. Many students were from departments other than CS, and had varying levels of programming experience. Over the course of the semester, I led lectures, optional tutorials, and personalized feedback sessions. At the end of the semester, even these students from outside of computer science were able to complete technologically sophisticated projects with immediate impact in their own fields, from a tool designed to assist in the early diagnosis of Alzheimer's, to an app designed to make it easier to find local cycling groups.

I was also able to supervise both a senior undergraduate (Tim Swast, currently at Google), and a masters student (Subhadip Ghosh, currently at Microsoft). Tim and I developed a tool for the digital humanities capable of showing word usage in an entire century of English texts. With Subhadip, I worked to improve the LayerCake tool for the visualization of viral mutations in HIV populations. These experiences developed my supervisory skills, but also the students' skills as both software engineers and as researchers.

Future Objectives

I intend to continue my commitments both to introducing new students to fundamental STEM ways of thinking, and interdisciplinary. I also hope to expand my mentorship and guidance of undergraduate and graduate students.

In addition to contributing to core classes in my department, I intend to develop specialized courses in visualization, and data science. Topics of special interest to me are *visual rhetoric* (the persuasive employment of visual images including visualizations), the *digital humanities*, and *statistical graphics*. In addition to being courses in important emerging topics, I believe that courses in these areas (and visualization generally) are useful places for fostering interdisciplinary collaboration, germinating longer-term research projects, and generating interesting research questions.