Teaching Statement

Ayaan M. Kazerouni

My teaching philosophy is grounded in *research* and *experience.* As a computing education researcher, I have carried out numerous studies assessing various educational approaches, and am comfortable evaluating and improving my own teaching practices based on theory and evidence. I have also been deeply involved with the third-year *Data Structures & Algorithms* course (CS 3114) at Virginia Tech. I have served as a teaching assistant for two semesters, the instructor of record for two semesters, and for the past five years I have consistently been involved in the planning and execution of several technical and non-technical aspects of the coursework.

Over time as an educator and a researcher, I have converged on a few guiding principles.

Cultivating motivation and engagement. Learning is most effective when the student is motivated and engaged. To drive engagement in class, I made heavy use of multimedia (e.g., interactive algorithm visualizations) and relatable analogies (e.g., the movie Inception as an analogy for recursion). To increase motivation, I made use of the MUSIC model of academic motivation [2]. It draws on research in psychology and education to model motivation as a combination of *eMpowerment*, *Usefulness*, *Success*, *Interest*, and *Caring*. I have promoted these components as an instructor for the Data Structures & Algorithms course at Virginia Tech. Students were *empowered* to self-regulate the pace at which they completed practice assignments, to choose partners for large projects, and to design their solutions to these projects. Taking a cue from my research, I emphasized to students that their *success* in programming projects (and indeed, the course) depended on their process (i.e., actions within their reach), and not on inherent qualities that they may not believe themselves to possess. This is particularly important in the computing discipline, which faces a crisis of diversity, leading many underrepresented groups to falsely believe that do not belong in the computing community. Finally, students appreciated my efforts to bolster their *interest* in the topics I was teaching, as well as the topics' *usefulness*:

> *Mr. Kazerouni always had lots of insightful observations and real-world examples to accompany his presentation material, which kept the lecture engaging.*
>
> *– Excerpt from end-of-term evaluation*

Effective use of formative assessments. I plan to make *formative assessments* a cornerstone of my teaching practice. Effectively used, formative assessment can lead to many positive outcomes for students [1]. Indeed, a driving hypothesis behind my research has been that formative feedback can help improve students' software development practices. I plan to extend this to my teaching as well. For example, I will *increase engagement* by using active learning strategies in the classroom; *provide emotional and cognitive support* by encouraging and supporting pair programming and group-based collaborative work; and develop students' ability to *learn how to learn* by incorporating strategies like peer instruction and encouraging students to take ownership over aspects of their learning.

I used formative feedback to great effect while teaching Data Structures & Algorithms at Virginia Tech. For example, while teaching about a given data structure, I encouraged students to *hypothesize* about its operations and their effects. We would then accept or reject hypotheses using interactive visualizations. Not only did this engage students in classroom discussion, but it also drew forth students' preconceptions and misconceptions, allowing me to actively tailor my teaching for the students at hand. I took efforts to ensure that participation was spread out over the entire classroom, and not just among the enthusiastic front-row students. I did this by learning students' names and engaging the class in dialogue, rather than specifically calling on students who seemed disengaged.

Mentoring. I strive to develop deeper relationships with students. I have played the role of *mentor* to my undergraduate students outside the classroom: they have sought my guidance regarding opportunities for undergraduate research, and their decisions about whether or not to pursue graduate education. I encourage these contacts beyond the classroom by maintaining an approachable and welcoming atmosphere in the classroom as well as during office hours.

I have also mentored current and future researchers. I worked with two master's students on a project in which they examined properties of student-written software tests; I helped them to define the project, prepare and analyze data, and interpret results. I am currently collaborating with a PhD student on a planned conference paper submission, and have guided him through the process of proposing the work, planning and executing interviews with student subjects, and preparing the manuscript.

Undergraduate research is important to me. Much of my desire to pursue a doctoral degree can be attributed to my undergraduate research experiences at the University of West Georgia. I worked on project areas as diverse as marketing, economics, and assistive technologies. I am eager to work with students to foster this same long-lasting intellectual curiosity.

Fostering diversity in the classroom. Perhaps the biggest push for diversity in computing can be made in the classroom. Research has shown that access to and interest in CS in high school is effective at getting underrepresented minorities interested in computing. As a professor, it is my responsibility to support at the college level students from all backgrounds, and particularly those who might not feel welcome in the computing community. It is crucial to empower and motivate them to *continue* to participate in computing. I will act on this particularly strongly in introductory computing courses. These courses typically have the lowest retention rates, disproportionately losing already-underrepresented demographics. I will tailor my teaching to use inclusive practices to foster a welcoming environment, ensuring that all students feel a sense of belonging in my classroom and in computing in general.

We must also help students to appreciate their responsibilities as future computing professionals. I will do this by situating computing in its broader context, using case studies and assignments to infuse issues of ethics and diversity into coursework. For example, when I taught Data Structures & Algorithms at Virginia Tech, I had students write essays about how the concepts we studied are applied in the real world, and their impact on the societal landscape. This also helped to contextualize the topics I taught.

Teaching Plans

My experiences thus far have prepared me to teach a range of undergraduate CS courses, including core CS topics, ethics in computing, software engineering, web and mobile development, databases, and user-interface development. I can teach graduate courses in computing education, software engineering, and software testing.

I look forward to entering a more professional teaching role as a faculty member, and am eager to educate and mentor the next generation of engineers, scientists, and scholars.

References

[1] Paul Black and Dylan Wiliam. Developing the theory of formative assessment. *Educational Assessment, Evaluation and Accountability*, 21(1):5–31, Feb 2009.

[2] Brett D Jones. Motivating students to engage in learning: The MUSIC model of academic motivation. *International Journal of Teaching and Learning in Higher Education*, 21(2):272–285, 2009.