Teaching Statement

Wenting Zheng

Teaching experience. I have served as a teaching assistant (TA) both as a master’s student at MIT, and as a doctoral student at UC Berkeley. At MIT, I was a TA for one semester of 6.033 (undergraduate computer systems) at MIT, helping lead tutorials on the class projects and holding office hours for students. At Berkeley, I was a TA for two semesters of CS168 (undergraduate networking). During the first semester, I taught three weekly sections. I was also in charge of developing the course material for one of the sections. During the second semester, I led the development of a new course project focused on network measurement. This was a great experience as I learned how to structure a class project so that the workload is reasonable, but also open-ended enough so that the students could explore and think more deeply about the problems. As part of the project, I held office hours for students and helped them with any questions they had.

My approach to teaching. Based on my experience both as a student and as a TA, I have found that one of the best ways to teach new knowledge is by actively engaging students through interactive activities. I believe that teaching is an activity that consists of more than just a professor lecturing students. It is essential that students actively participate in the learning process as well.

For undergraduate courses, I would like to design lectures that involve live demonstrations and interactive activities. For example, when I was a TA for CS168 under Professor Scott Shenker, one of the activities that he designed was a demonstration in which students simulated a routing protocol using bean bags. I observed that, because the demonstration had captured every student’s attention, the entire class remained engaged throughout the entire lecture. While including this type of activity in every lecture is likely impractical, systems and security courses do tend to have more opportunities for interactive demonstrations because many fundamental but abstract concepts can be found in real world applications.

Sections and homework assignments are also important tools for reinforcing the concepts taught in lectures. Since students absorb information at different rates, it is possible that a small portion of the students will not fully understand every concept that was taught in lectures. Even if a student thinks that s/he understands the concepts, they may still have gaps in their understanding. Therefore, I would like to design sections to focus on problem solving because I believe that working on concrete problems is the best way for students to learn. However, since participation in a large class (or even a large section) may be intimidating for some of the students, I plan to structure section discussions to be conducted in small groups, so that all students can feel comfortable enough to participate.

In terms of homework assignments, I believe that hands-on projects lasting approximately two to four weeks are the best way to teach students about the practical applications of concepts learned in class. When possible, I hope to structure projects to be cumulative in order to give students the opportunity to build a holistic system over the course of a semester.

Graduate classes and seminars are different from undergraduate classes because students have already mastered the basics and are now learning about cutting-edge research. I plan to structure the classes primarily around reading and discussing papers. For homework, each student will write up a review of each paper. In class, students will discuss the reviews program-committee style. Since I find that younger students tend to be more critical of the papers that they read, I plan to discuss both the positive aspects and the shortcomings of these papers.

Courses I would like to teach. Given my previous teaching and research experiences, I would like to teach core undergraduate- and graduate- level courses in operating systems and security. I would also like to teach a seminar on system security that covers secure collaborative computation, including topics such as hardware enclaves, secure multi-party computation, and secure training and inference systems.

I would also like to teach a special seminar for first-year graduate students on how to navigate the research process. When I was just starting graduate school, I had difficulty understanding various aspects of research, such as how to find a research problem and how to write a good research paper. This seminar would cover topics such as what research problems to tackle, how to write good paper introductions, how to deal with paper rejections and the ethical implications of research.

Finally, I would like to organize a similar seminar for undergraduate researchers. During the past year, I helped organize DARE, the first program at Berkeley that matches undergraduate researchers
to research opportunities with a special focus on diversity and inclusion. I would like to continue this program once I become a professor, and as part of the program would include leading a seminar focusing on how to do research as an undergraduate student. I believe that it is important to get as many students as possible interested in research and educated about the research process early on in their undergraduate careers so that students of all backgrounds have the opportunity to see research in computer science as a viable career option for them.

**Research mentoring.** I was fortunate to have advised seven talented undergraduate researchers throughout my time in graduate school. My approach to advising new undergraduate students is to first match them to an existing research project. One technique I find useful is to carve out a concrete and well-defined piece of the research project for each student to work on. Ideally, this piece of the project would span both design and implementation so that the students have the opportunity to work independently and take ownership of their contributions. I have weekly project meetings with all of my undergraduate students, and, should they feel the need to get more help, schedule impromptu meetings as necessary.

For example, I am currently working with an undergraduate researcher, Ryan Deng, on the Cerebro project. Ryan did not know much about cryptography when he first started working on the project, but was able to quickly catch up on the basics by reading the related work I provided to him. I also gave him a specific piece of the project (cost modeling of cryptographic protocols) to start with. This piece of the project was a good fit for Ryan because he could think about the design (how to derive the cost model equations) as well as the implementation (running experiments and measuring the costs). Ryan participated in the project meetings and I always allocated time for him to discuss his progress and any questions he might have. I also set up extra meetings with him if he felt that it was necessary. Throughout this process, Ryan has grown into an independent researcher. He is now starting a new project, on which he will be the lead student researcher, and is also applying to graduate school to continue research.

In general, I believe that it is important to cultivate undergraduate students to become independent researchers, but also to be available to help whenever they encounter problems. That said, each advisee is different, and it is essential to work with each student individually to find the advising approach that is most appropriate for them given their personal strengths, weaknesses, and goals. I will always be available for my future graduate student advisees and will encourage them to communicate with me whenever they run into troubles. Research is a difficult path and can sometimes feel isolating, so it is my job as an advisor to give my students as much support as they need to be successful and happy in their graduate careers.