Masoud Moshref Javadi - Teaching Statement

Teaching and advising are essential and exciting parts of an academic career. I can teach networking and operating systems along with introductory programming courses in undergraduate level and present software-defined networking, cloud computing and distributed systems in graduate level. I look forward to mentoring students to learn and practice critical thinking, problem-solving and presentation skills.

I love learning, and I enjoy helping others experience the learning process. That is why I started teaching back in my undergraduate study in Sharif University of Technology where I was TA for many courses. An exceptional experience was the Computer Workshop lab for freshmen that is taught completely by senior and graduate students under the supervision of a faculty and a head TA. I presented for two semesters and was the head TA once. In USC during my Ph.D., I was TA for undergraduate Operating Systems (OS) course that involved a major programming project. I presented lectures for guiding 70 students through the project, held office hours and graded the project assignment. I have also been the guest lecturer for two courses and advised undergraduate and graduate students. In the following, I describe my teaching philosophy to facilitate learning and my goals in course development.

Learning by doing: I believe students will be more excited to learn about a topic once they use it in experience even in an emulated environment. When something does not work in the experience, students have to refer back to theories and techniques they may have overlooked during the lecture. Even better, once students work on real systems (many available open-source), they learn the complexities of actual systems, an experience that is also valued in the industry. When I was TA for the OS course, I held office hours to answer students question for a project on Pintos (an instructional OS). I had to review different concepts such as synchronization and memory management for students to help them finish the project.

Learning by immediate feedback: Learning is not possible without feedback. There are many ways to reduce the time to give feedback to students: (a) For the hands-on projects, the public test cases uncover (sometimes trivial) mistakes for students in an instance. Thus, they do not need to wait until office hours. (b) For many of the hands-on experiences, it is possible to make a competition. I remember I had a cloud computing course in USC, and the goal was to implement the fastest distributed system to process texts and photos in a photo-sharing service. Being able to compare the performance of my implementation with other groups was a continuous motivation. (c) I found the online discussion forums (e.g., Piazza) very helpful for students to get answers on recurring questions fast. In addition, they provide a platform for students to share knowledge.

Learning through research in graduate courses: I believe graduate courses should be designed around research through different practices: (a) Reviewing basic (sometimes old) papers because in computer science old ideas come up again and again in new contexts. I was thrilled during Advanced Operating Systems course in USC to know how much designing a fast remote procedure call has common with new techniques for bypassing network stack. (b) Reviewing state-of-the-art papers by a presentation from authors or guest lecturers. I have been the guest lecturer in two courses in USC describing my work and a topic in my research. I enjoyed the experience that students could ask directly about the process of coming up with the idea, the research challenges and future work, the opportunity that is not possible just by reading papers. (c) Performing a research project on a cutting edge problem but in small scale. This provides a taste of what happens in research...
laboratories, but it can result in a publication (most likely a workshop paper) in a semester. I have advised a few graduate students on their course projects, which ended up with a workshop paper. Because the project was small, two graduate students could finish it during a semester and even have enough time to make mistakes and try different solutions.

**Course development:** I am looking forward to presenting courses on special topics such as cloud computing, datacenter networks, distributed systems, optimization in networking and software-defined networking for graduate level. Especially, I will pitch the courses around real-world networking problems when computing happens at large scale with low delay and high availability requirements. I show how to abstract a problem to apply theory and how to make it practical using system building techniques. I also want students to read papers about operational networking systems in the experience track of conferences in order to understand how research ideas are realized in operation and what challenges happen in deploying the systems. For undergraduate level, I can teach courses on networks, operating systems, data structures and introductory programming courses.