Teaching Statement of Abhigyan Sharma

"Mind is not a vessel to be filled, but a fire to be kindled." As a teacher, I strongly believe in this quote by Plutarch. I teach to nourish the curiosity and intellect of my students while orienting them towards the topic of my teaching. I emphasize to them that a textbook describes a finite body of knowledge but the endeavor of knowledge and learning can and should continue beyond the walls of the classroom and beyond the end of their college careers. On my part, I teach with a realization that my students are the ones whose work is going to be a part of the textbook that the next generation of students will read.

I am passionate about teaching computer systems and networks since they are indispensable part of modern economy and society. The core principles taught in these courses such as mutual exclusion, indirection, virtualization, caching and scheduling, have a remarkable reuse in the design of numerous applications and services. A clear understanding of the principles of designing computer systems is necessary for any student of computer science, independent of the sub-area that they work in.

My teaching experience comes from various sources. I was a teaching assistant for an undergraduate course on Digital Forensics, in which I held weekly discussion sessions with students regarding their assignments, assisted students during lab sessions, prepared in-class quizzes, and graded exams. I have given presentations at prestigious conferences, which has taught me how to present deep technical concepts in an accessible manner to a general audience. I have mentored junior graduate students on projects done in collaboration with them. I have led class discussion on several occasions in seminar courses on Big Data, Green Computing, and Content-Oriented Networking as well as in reading group meetings on systems and networking. I have given guest lectures on my research at Indian Institute of Technology Kharagpur and at Indian Institute of Technology Patna that were well received by faculty and students in those departments.

I have found that students learn best when they can relate the new material with their existing knowledge. I make an effort to build these relations in several ways. I motivate a problem by relating a new problem to the one they may already know. For example, routing problems in a networking course are often related to graph algorithms that many students would have learned in an earlier algorithms course. I have observed that when new ideas seem to appear unexpectedly during a lecture, students tend to lose focus. I strive to keep a logical succession of ideas that encourages students' interest and helps them to think and to question why a particular approach was chosen at each step. I add backward and forward pointers to a related material that is covered earlier or later in the course, or could be covered in another course, to help students form a mental web of related concepts, and understand the broader applicability of an idea, e.g., caching can be used inside a processer, by the operating system or by a distributed storage system.

I have enjoyed mentoring junior graduate students. Aditya Mishra and Vikas Kumar collaborated on one of my early papers and Vijay Pasikanti collaborated on the Shrink paper. In these experiences, I have discovered two important facets of advising. First, an advisor should provide an optimum level of guidance to students. Too much guidance can be stultifying for students and conversely too less of it can be puzzling. While working with Vijay, I first erred on the side of caution by guiding him at each step. I found out that Vijay loved to find his own way to solve problems. Later, I found it effective to convey the higher order bits of the problem and letting him find solutions to the extent possible. This approach proved to be productive and Vijay made a substantial contribution to the project by designing and implementing one of the modules in Shrink, mostly by himself. Second, an advisor should be keen to listen to and learn from students. As I have experienced in my interactions with my advisors, this attitude gives students confidence in their ideas, so that they mature as independent thinkers and are able to drive their own research.

I am interested in teaching courses in networking, operating systems and distributed systems, introductory courses in data structures and programming, and a seminar course on convergent network and content delivery infrastructure. I also plan to design a graduate-level cross-disciplinary course focusing on datacenters, the mega-engines that drive modern day computing. The course would cover software, networking, electrical and cooling technologies, and would include material on currently deployed technologies as well as cutting-edge research ideas. The cross-disciplinary nature of this course would offer a better understanding of how the different sub-systems in datacenters influence each other's design.

I plan to enhance my teaching by bringing in elements of my research into it. First, I would keep an experimental component in my courses that uses the state-of-the-art educational tools (e.g., Mininet, NachOS) and testbeds (e.g., Emulab, PlanetLab) wherever necessary, allowing students to perform mini-

experiments to test their code. These experiments would stress the importance of implementing ideas into working prototypes and performing experiments with it to test the practicality of an idea. Second, I would offer open-ended course projects in my graduate-level courses through which students can explore new research ideas. My dissertation research started as a course project with my advisor, and through my teaching, I want to offer my students a sound platform on which their ideas take flight.