

## Teaching Statement for Dilsun Kaynar

Throughout my PhD studies and postdoctoral research I have taken several opportunities to teach courses and to supervise undergraduate and graduate research projects. I find it enjoyable and rewarding to interact with students, and this is a key factor in my wish to pursue an academic career.

My thoughts about teaching computer science have been shaped through my experience as a tutor (which corresponds to a teaching assistant), and as a lecturer at the University of Edinburgh. I worked as a tutor for courses that introduced basic concepts of computer science and software engineering, taught programming in Standard ML and Java, and techniques for reasoning about algorithms and programs. My students consisted of young undergraduates who were exposed to computer science as a scientific discipline for the first time in their lives. I then became the lecturer of a course on distributed systems that covered major design issues regarding communication, fault-tolerance, security, and included an introduction to distributed algorithms. I was responsible for the preparation of the entire course including the choice of the textbooks, and preparation of the syllabus, teaching materials, assignments, projects, and exams. In contrast to my prior teaching experience, which involved close interactions with a relatively small number of juniors, this course had a large number of students consisting of seniors and Master's students who were well-informed about several aspects of computer science.

I observed that teaching introductory level courses presents us with a unique opportunity to influence the way the students approach computational problems, to help them obtain a robust basis of knowledge, and to instill some working habits that will most likely remain with them in the rest of their career. I find it particularly rewarding to observe how juniors evolve as a course progresses. In advanced courses we can inspire students to apply their knowledge and skills in various creative ways. Such courses can also be inspiring for our own research; since we deepen our own understanding about a matter in trying to explain it, and in interacting with others.

In the process of teaching, I learned how important it is for a teacher to be prepared for a class, regardless of how knowledgeable the teacher is about the subject matter. To be clear and effective, one has to organize the material carefully to ensure a natural flow and to allow for some light-hearted moments to revive any lost attention in the class. In preparing for a class, I do not hesitate to spend a good amount of time to find the right examples. In addition to classic examples that have proved to be powerful in conveying a concept or an idea, whenever applicable, I try to find timely examples that relate to students' daily lives or newly emerging technologies. In a fast-moving field such as computer science, it is essential for a teacher to be able to adapt the teaching material to address the recent developments in the field and to help students to place the subject matter in the right context.

The preparation of assignments and projects is also of great importance since these are essential tools for urging students to do the mental and practical exercises required for learning. They must be carefully crafted to teach and assess at the same time. I have observed through my own teaching experience that some students who are typically quiet may become more active participants in the class after doing an assignment due to the improvement in their understanding of the subject. My experience also taught me that there will always be some students who master the course material more easily than others, and it is useful to challenge these students at least with some parts of an assignment or a project to keep them interested in the course material. The engagement of such students in the course material makes the lectures more interactive and creates a better environment for learning. I once set an easy programming project due to my own lack of time and some students were apparently critical because I had underestimated their capabilities. From then on, I paid greater attention to adjusting the difficulty level of such projects.

During my postdoctoral research I have supervised several undergraduate research students and Master's students at MIT, on projects related to building software tools for simulation and verification of distributed algorithms based on mathematical I/O automaton models. I think that an important duty of a supervisor is trying to make himself or herself available for spontaneous discussions and feedback, in addition to holding regular meetings with research students. This approach paves the way for more productive research. In working with my supervisees I try to make them realize that a research project is more than just solving a problem or implementing a system. They have to be able to explain what they are doing and why they are doing it that way. Many research students are initially intimidated by the prospect of writing a paper, or a thesis. I encourage them to write about their work and document their progress from the very beginning of their research.

I recognize that it takes a lot of effort to become a competent educator. I think that my prior teaching experience will help me in teaching core computer science courses and in developing new ones, but I also look forward to improving my skills as a teacher and an organizer through related career development programs.