Teaching Statement

Shoaib Kamil
skamil@mit.edu

It is an enormous privilege to be able to teach and advise students. I have been incredibly lucky to have already engaged in teaching and mentoring for a number of years, but I am especially excited by the opportunity to do so as a professor.

I have taught courses (as a lecturer and teaching assistant) in a wide range of areas of computer science, including compilers, software engineering, parallel computing, and algorithms. As a professor, I can teach basic principles of programming and software engineering for undergraduates, parallel computing and optimization, compilers, as well as courses in program analysis and software verification.

Approach to Teaching

My approach to teaching is to cater to a combination of learning styles while diligently combining theory with practice. I present theoretical concepts and principles but attempt to always combine them with concrete examples that demonstrate important aspects of theory but still connect with student experiences. For example, when describing basic fork-join parallelism while teaching UC Berkeley's graduate *Applications of Parallel Computing* course, I walked through parallelizing a simplified version of a project the students had just completed, starting with animations that showed the basic ideas, and then the simplest implementation. Exercises like these help students bridge between conceptual and practical knowledge.

To effectively reach all students in each course, I believe lectures must combine different learning styles, caterint to visual, auditory, and practical learners. I strive to include all three in each lecture. In addition, I have found that giving students multiple opportunities for self-assessment of learning during a lecture leads to interesting and useful questions and discussion. For example, the lectures for *Software Engineering* at Berkeley consisted of 10-12 minute "chunks," and after each chunk students are asked self-assessment questions which they discuss with their neighbors. I found this structure enabled students to figure out what questions they need to have answered in order to understand the material. As a result, more students understood more of the material, which was reflected in overall student performance and in course evaluations.

Outside of lecture, I am passionate about helping students who are trying hard but not quite succeeding to the fullest extent. In many cases, these students benefit from even small amounts of one-on-one time, where I can cater an explanation to take advantage of what the particular student understands well. In every course I've taught, I've been heartened to see the "click" when all a student needed was an explanation from a different angle.

Mentorship/Advising

One of the most fulfilling experiences for me is to see the personal and professional growth of a student over the long term. I'm very excited for the opportunity to work with students for the entire arc of their academic journey. In my current position, I've been privileged to supervise several undergraduate researchers and to help supervise masters and PhD students. While at UC Berkeley, I was involved in mentoring students from underprivileged backgrounds while they conducted summer research, as well as working with final-year undergraduates and fifth-year masters students. These were all immensely successful, both in terms of publications (four conference papers just from undergraduates) and in terms of what students went on to accomplish afterwards.

For me, the most challenging aspect of mentoring students is figuring out the interaction style and meeting frequency for each student. Every student is different. I have found that the best strategy is to adapt to each student's needs by learning what approach works best as we work together— but always being available if students are stuck.

Teaching Interests

My teaching interests are varied, both because of my personal interests in the intersections between computer science and other subjects, and because of the different areas my research encompasses. I'm especially interested in teaching *Programming Languages & Programming Language Design*, *Compilers*, *Program Analysis*, and *Parallel Programming*, among others, at both graduate and undergraduate levels.

I'm particularly interested in integrating software engineering concepts into any course I teach. As the demand for computer science courses continues to increase, it is important to emphasize solutions to the real-world problems programmers encounter. I want to ensure students in my courses follow best practices that will be useful in whatever they do after graduation. Similarly, with parallelism becoming more and more ubiquitous, I want to integrate questions of parallelism into each course, so students better understand the particular problems that come with moving from serial execution.

Summary

I am excited for the opportunity to teach computer science, especially given how important computing is becoming to every branch of science and study. Computer science education is something all students can benefit from. When it comes to advising and mentoring, I'm passionate about helping students develop their own interests and voice. I look forward to the opportunity to both teach and advise students who will use programming and computer science to build the future.