Teaching Statement

Samuel Madden
Computer Science Division
University of California, Berkeley
madden@cs.berkeley.edu

A desire to teach is one of the primary reasons I am seeking a faculty position. My own experience with outstanding faculty, both in large undergraduate classes and through interactions from research and advising, have been crucial to my own career. Someday, I hope that one of my students will write the same of me.

My research experience qualifies me to teach undergraduate courses in databases, operating systems, software engineering, and networking, as well as introductory programming and data structures courses. At the graduate level, I would like to teach courses in operating systems, distributed systems, or advanced database systems.

Beyond core courses such as these, I would like to teach graduate level seminar courses exploring current research in databases and systems: for example, I believe my experience would allow me to offer a unique perspective on courses in adaptive query processing or sensor networks. At the undergraduate level, I would like to involve students in projects related to current topics in computer science – for example, by offering a ubiquitous computing class where students are asked to devise and implement a project using sensor hardware and software, such as Mica motes and TinyOS from Berkeley.

Teaching Experience and Philosophy

At Berkeley, I was the sole TA for two semesters of CS169, the undergraduate software engineering course. Each semester, the course had approximately 50 students, and I was responsible for teaching sections and organizing and supervising student projects, which were conducted in groups of 5-7 students. I was also responsible for maintaining the computing and software infrastructure for the class and assisting with grading.

As a TA, I became a firm believer in the philosophy of teaching via real-world example. Software engineering, if taught the wrong way, can be dry, but when accompanied by real-world accounts of software engineering failures – such as the Therac-25 software, for example, or by accounts from personal research projects, such as my own experience building the initial Telegraph query processing system, which was ultimately scrapped due to poor performance and a bloated code base, can lead to lively discussions and give students a real-world perspective on otherwise abstract problems.

My own experience at MIT and Berkeley convinced me that another key aspect of teaching at both the undergraduate and graduate level is requiring students to engineer real systems. Although simulators and toy systems can help students focus on coding small algorithms and understanding concepts, I believe students will not learn to be good programmers or researchers unless they are required to engineer substantial software artifacts from scratch.

Mentoring Experience

Finally, during my four years at Berkeley, I have supervised five undergraduates and worked closely with several others. Several of those students are now on their way to graduate school, and, in one case, I worked with an undergraduate to submit a paper to a workshop on sensor networks (IPSN 2003). I have also served as a mentor for other graduate students in the group, helping them to refine their ideas and improve their writing.

This work with younger students is one of the primary reasons I decided to pursue a faculty position – industry simply does not provide the opportunities to teach and be taught by close, one-on-one relationships with smart and enthusiastic students.