TEACHING STATEMENT

Baishakhi Ray

One of the primary reasons of my interest in academia is the opportunity to teach students and to produce new leaders in computing. I believe that teaching not only helps students to learn new concepts but also enriches the teacher to have a more coherent understanding of the topics. Teaching also inspires to build a vibrant research environment by passing on ideas and passion of inquiry to students. As a teacher, I truly believe:

The art of teaching is the art of assisting discovery. (Mark Van Doren)

Teaching

My experience in teaching dates back to my undergraduate days, when I tutored high school students to prepare for their college entrance exams in India. More recently, I have guest-lectured in several software engineering courses at the University of Texas at Austin and University of California Davis. I have also been a teaching assistant and a grader for courses on operating systems and networking during my graduate school career.

As most of my research as well as teaching experience is in the experimental aspects of computer science, I have found a hands-on approach is the most effective in preparing a student towards a successful engineering career. My teaching philosophy consists of three key factors:

1. Employ a hands-on approach: I think learning is most effective, when one applies the learned concept in real life. Hence, I strongly encourage implementation in order to grasp new concepts. I prefer to incite and distill the solutions from the students' thoughts over telling them the solutions straightaway. In this process, students not only get to know the solution, but also can identify alternative approaches to solve the problems by themselves. For example, while teaching the concept of unit testing in an undergraduate class, I first showed the students a buggy program and asked them to generate inputs that will trigger the bug. While coming up with the solution, they have already written their first unit test case. In fact, I have noticed that such an approach helps the students to think independently and thus motivate them in active research.

2. Show the bigger perspective: While teaching, I also make sure to present a bigger perspective of the problem before introducing a particular solution. For example, while teaching `Git` version control systems (VCS) in an undergraduate class, first I explained *why we need a distributed VCS like Git over traditional VCS like SVN* before getting into the details of Git commands. I further demonstrated scenarios where Git outperforms SVN. I believe such approach also helps students understanding the relationship between abstraction and its implementation.

3. Encourage active participation: Finally, I strongly encourage students to actively communicate in my classes. Instead of long lectures about a topic, I prefer to make the discussion interactive by involving the students. This not only helps the students feel more confident by answering simpler questions, but also allows advanced students to ask about related topics of their interest and enriches the overall learning.

Given my background in systems and software engineering, I can teach undergraduate classes in *software engineering*, *basic programming courses*, *data-structure*, *program analysis*, *networking*, and *operating systems*. At the graduate level, I would like to teach more software engineering courses on *software evolution* and *empirical software engineering*. I would also like to design a new course on upcoming technologies. For example, I plan to teach a senior undergraduate class on *mobile application development* where students will create applications using Android/iOS SDKs. I will encourage students to work in groups and use Git to collaborate. In this way, students will embrace cutting edge technologies as well as learn important software engineering tools.

Mentoring

Mentoring is challenging, especially because students are very different from one another in terms of research taste, work patterns, and background knowledge. As an adviser, it will be my responsibility to be inclusive of such diversity and help my students to fulfill their potential.

Throughout my research career, I have been fortunate to co-advise many undergraduate and graduate students as well as a high-school student for a science fair. In the process, I realized that students at different stages in their career need guidance in different ways. For example, the beginning students need closer attention. I help such students in choosing interesting research problems as well as assist them with low-level technical issues. For example, I extensively helped Casey Casalnuovo, a beginning graduate student at University of California Davis (UCD), to conduct an empirical study on the use of assertions in source code. From asking the relevant research questions, building statistical models to writing a technical paper, Casey worked under by supervision. I guided Saheel Godhane, a beginning graduate student at UCD, to write control and dataflow analyses using LLVM/Clang infrastructure. I also mentored Tyler McDonnell, an undergraduate at the University of Texas at Austin, to study API Stability and Adoption in the Android Ecosystem. I designed the study and mentored Tyler to conduct the experiments. I further helped him to write a technical paper, which was published in International Conference on Software Maintenance, 2013 (ICSM). As the students progress through their research projects, become comfortable, and start taking ownership of their projects, I helped the students with the high level details, positioning the project, writing the paper, etc. Following this path, I and Vincent Hellendoorn, a visiting graduate student at UCD, had a very successful collaboration experience, and we started working on several paper submissions together. My mentoring have resulted in successful research papers, undergraduate and masters theses. Taariq Chasmawala, the high school student whom I mentored at the University of Texas at Austin, won awards from Intel and the U.S. Army for our work. Taariq ended up joining Computer Science undergraduate program at the University of Texas at Austin.

I also had the opportunity to encourage female undergraduate students in research during my time as Vice President of Graduate Women in Engineering at the University of Texas at Austin. I held several poster sessions and seminars to familiarize the undergraduates to ongoing graduate research projects in the Electrical and Computer Engineering department. I plan to continue inspiring undergraduate women to pursue graduate school and career in computing, in general.

As an adviser, I will encourage my students to conduct independent research while guiding them to produce quality work. I will try to hone their faculty by organizing reading groups, encouraging collaborative research, and challenging them to their best. Last but not least, I want my students to be excited about their research, and I will try to harbor a friendly erudite environment to achieve the same.