Teaching Statement

Janki Bhimani
bhimani@ece.neu.edu

I am thrilled by the opportunity to fully integrate teaching and mentoring into my daily routine. Not only do I consider these activities to be vital to my passion as a scholar, but I find them to be among the most impactful and personally rewarding. I will describe how my past decade of teaching in three diverse formats—traditional lectures, hands-on lab sessions, and one-on-one tutoring, have honed my teaching philosophy. I will then conclude with a discussion of potential courses that I look forward to teaching.

Teaching Philosophy

As an instructor, I realized that my responsibility should go beyond just conveying knowledge. It is more important to motivate students to acquire and apply knowledge, teach them critical thinking skills to solve newly occurring problems in their fields of study, assistant them to build career foundations and prepare them to succeed after graduation. Based on my experience, the following three pillars construct the strong base of my teaching philosophy:

Be Rigorous And Engaging.
I strongly believe that the quality of being extremely thorough and accurate in basic foundation principles is important in teaching. I will emphasize to students that the specifics of all the programming languages, libraries, frameworks, and deployment services that are in vogue at the moment, can change at a rapid pace. All new technologies are easy to learn once students have developed strong foundations. I have noticed that students take a course more seriously when the instructor is firmly committed to rigor and high standards. Apart from technical depth of content, I will also encourage everyone in the classroom to express their thoughts. To bring technical subjects to life, I would like to teach using the combination of various techniques such as, white board discussions, power point presentations and audio-visual illustrations to demonstrate live operations. The best positive comment that I have received from my students is that they can always concentrate and feel that classes are shorter than actual, when I teach.

Think More Than Memorize.
It is my belief that students learn more by thinking than by memorizing. Therefore, in my discussion lectures, I do not spell out the solutions in a snap. Instead, I will design step-by-step sub-questions to guide the students through the problem, and encourage them to come up with the intended solution or alternative ones. I will also ask the students to compare and explain the advantages and disadvantages of different solutions to the same problem. When students come to my office hours asking questions, I usually answer by asking them questions that lead them toward the complete solutions. Although this method may take extra time, I find it is very effective in helping students learn their skills and build their confidence.

Be The Learner Forever.
As Phil Collins rightly said "*In learning you will teach, and in teaching you will learn.*" As computer science and engineering is a growing and ever-changing field, I believe teaching is not only about transmitting knowledge to students, but also about inspiring students to inquire and learn independently. It is a process of collective improvement for both students and the teacher. Each student is unique in their personality and working style. I have learned and will continue to tailor my mentoring style for bringing out the best of each student. I want to become a professor because I am enthusiastic to engage in this exciting process, fostering and educating the best young talents of our next generation.

Teaching Experience

I feel very fortunate that I have had many tutoring and teaching experiences. I independently taught a standard 4-credit course named Fundamentals of Engineering Algorithms (EECE 2560) in Fall 2017 semester. This course covers the design and implementation of algorithms to solve engineering problems using a high-level programming language. It reviews elementary data structures, such as arrays, stacks, queues, and lists, and introduces more advanced structures, such as trees and graphs and the use of recursion. It incorporates manipulating these data structures as well as their use in problem solving. It also introduces algorithm complexity analysis and its application to develop efficient algorithms and emphasizes the importance of software engineering principles.

Fundamentals of Engineering Algorithms is a challenging class to teach because topics covered in it range from basic to advance techniques of algorithms. It is tough to teach because this class includes students from various

majors and background. For example, in my class, there were 16 students with their major in various disciplines such as Chemical, Linguistics, Mathematics etc. My class included women, Black or African American, and Hispanic of any race. In this course, I delivered three 65-minute presentations weekly. I utilized engaging teaching techniques that include combination of white board discussions, power point presentations and audio-visual illustrations. I designed 5 hands-on projects, that were given as 10 homeworks equally scattered and in sync with class offerings throughout the term. I conducted a number of both ungraded and graded quizzes and exams. To give a better vision of how class topics can be practically applied, I initiated discussions on recent, evolving technological advancements and gave real-world examples where the algorithms they are going to learn have already been used. I encouraged students to think openly about how to improve existing algorithms, and motivated participation in class discussions by awarding them with "be bold" points. The students' feedback for "instructors' overall effectiveness" was a distribution with mean of 4.4/5 and median of 5/5, where 5 stands for "Always Very Effective". The main comments that I received are, *"Professor Bhimani demonstrated incredible skill in teaching this course."*, *"She is an Excellent teacher and mentor!"*, *"Definitely, I want to take another course if she teaches something else, because I learned a lot with her"*. Many of my students were successful to get Co-op and Internship opportunities after taking my course. One among them also got full time offer from Google to work as a Software Engineer. I feel very joyful with their success and I am proud of my students.

In addition, during my graduate studies I have served as a teaching assistant (TA) for several courses, including High-Performance Computing (EECE 5640), Fundamentals of Computer Engineering (EECE 7205), ST: Simulation and Performance Evaluation (EECE 5698) and Algorithms (CS 5800). My responsibility as teaching assistant was to advise students to work on projects and assignments using the knowledge learned in class. One-on-one tutoring was one of the major part of being a TA. I held office hours to provide hand-on guidance and answer specific questions of students. I prepared additional handouts for each lecture for students to gain in-depth knowledge on topic of discussion. I also substituted class for professor to conduct lectures. Moreover, while doing my undergraduate studies, I taught a course on Robotics Programming and Circuit for three semesters. I also conducted hands-on lab sessions to build whole robot including manufacturing chassis, designing and soldering circuit and programming controller and processor. I individually mentored many students to participate in world-wide robotics competitions.

All these valuable experiences, especially when held in conjunction with my research activities, led me to develop the following teaching philosophy that I plan to use to shape my courses.

Potential Courses

My interest and expertise in a broad range of domains and inter-disciplinary subjects will enable me to efficiently teach a wide variety of courses in computer science, information technology, computer systems, electrical and computer engineering, and data science. In particular, I am interested in teaching advanced courses in the field of parallel, concurrent, and multithreaded programming, high performance computing, big-data processing, cloud computing, distributed systems, and programming for data science. In addition, I would like to also teach entry-level courses, such as algorithms, programming languages, data structures, databases, computer organization, operating systems, robotics, logic and computation, and machine learning. Moreover, with my many internship experience in industrial research labs, I expect to contribute to the development of curriculum by creating advanced graduate courses on performance engineering, memory and storage technologies, and data management. I envision theses courses could contain topics on a holistic introduction to a list of performance profiling tools, trace the lifetime of storage devices and generate reliability analysis, measuring cache and memory latencies using synthetic workloads, studying the performance scaling nature, monitoring the memory and storage performance. Students will feel the great potential and usefulness of what they study. They will gain insight into both theoretical and practical aspects of these fields. Course materials will be selected among recent research articles and newly published books.

In summary, my industrial experience, academic background, teaching experience and research work make me well prepared to teach introductory as well as more advanced courses. I also intend to keep refining my teaching techniques with carefully planned, effective teaching activities and feedback from my colleagues and students.