After the day’s lecture I proposed the following question to the students: Suppose you are given a million integers to sort. What algorithm would you use to sort them and how much time and memory would it consume? This is an example of an interview question that I gave students enrolled in an algorithms and data structures course. We had just covered basic sorting techniques along with more specialized methods such as radix and counting sort. At the beginning of each lecture we discuss possible solutions to the question posed last time. The point of these questions is to encourage students to apply the concepts learned in school to tangible problems, allow them to practice communicating their solutions in a clear and concise manner, and help prepare them for future engineering opportunities.

My responsibility as an educator is to teach students how to think critically. I believe in adopting and developing innovative teaching techniques for conveying computer science and engineering concepts. Upon completing a course, students tend to forget the material after a short period of time. However, when students discover concepts in the course themselves their retention of the material is greatly improved. When I teach, my aim is to create an active learning environment. In computer science and engineering, an active learning environment is one that stimulates the students to attempt a solution based on their present experience and knowledge of the subject. I have been inspired and influenced by the work of George Pólya in education and routinely incorporate the advice presented in “How to Solve It” into my teaching. I also agree with Richard Feynman’s observation that students are often the source of new research.

Teaching Experience

During a lecture I strive to have the students visually understand the content, devise and carry out a plan to solve a problem, and ask if their solutions can be further refined. I prompt the students with appropriate questions and permit the lecture to be guided by their questions. Below are brief descriptions of my experiences as a teaching assistant and substitute instructor for various subjects at the University of Minnesota. The breadth and depth of my teaching know-how has been improved by each of these courses.

Artificial Intelligence. I’ve been a teaching assistant for both an undergraduate and graduate course in Artificial Intelligence. For the undergraduate course, I worked on a team to grade and prepare homework solutions, answer questions, and assess final projects for over 150 students. My first teaching experience was at the graduate level where I covered various search algorithms. I was also responsible for grading and preparing solutions to written assignments, and evaluating programming assignments.

Robotics. Introduction to Intelligent Robotic Systems is a graduate course that covers transformations, kinematics, inverse kinematics, dynamics, and control. In addition to my teaching assistant duties, I covered the lectures on inverse kinematics, a particularly tough area to grasp as a student and teach as an instructor. I also led the lectures on robot programming including demonstrating the Robot Operating System (ROS) for cloud-based robotic applications.
Computer Vision. The Computer Vision course introduces graduate students to perspective transformations, edge detection, image segmentation, feature tracking, and more. As the teaching assistant, I prepared assignment solutions and graded successive programming projects that culminated in a simple industrial inspection machine vision system. Additionally, I taught the material on 3D perception and introduced the students to the concepts behind stereo vision.

Algorithms and Data Structures. I’ve been involved as a teaching assistant and recitation leader for the honors and undergraduate Algorithms and Data Structures courses. In the honors course, I covered additional material during the recitations according to the interests of the students. In the undergraduate course, I taught back-to-back recitations for a class of approximately 80 students. For each course, I created my own lecture notes that supplemented the course material and provided additional exercises.

In each of the above courses I received excellent evaluations from the students and positive comments on my ability to explain difficult ideas and provide concrete examples.

Mentoring
At the University of Minnesota I’ve had the opportunity to mentor many high-achieving undergraduate students. Each semester I usually advise two to three students on various projects in robotics and computer vision. I have also supervised two undergraduates for their senior design projects in electrical engineering and computer science. By working with me, students have gained experience conducting research and our collaborations have led to publishable results.

Training
I have taken advantage of the excellent resources provided by the University of Minnesota’s Center for Educational Innovation. This includes workshops for first time teachers such as adapting a syllabus, designing engaging class sessions, teaching with technology, and creating effective assignments. I’ve completed training that develops strategies for helping students enhance their metacognition. I have also participated in workshops held by the Center for Writing with the goal of learning how to improve student writing.

Teaching Interests
My experience has prepared me to teach a wide range of classes at the undergraduate and graduate levels. At the undergraduate level, in addition to algorithms and data structures, I am prepared to teach courses in discrete mathematics and computational linear algebra. I’m also well-prepared to be an instructor for graduate courses in artificial intelligence, computer vision, and robotics. Furthermore, I am enthusiastic about the possibility of introducing these courses to a computer science and engineering program.