My primary goal as an educator is to help students excel in their careers. Toward that goal, I aim to develop their critical thinking, to build their foundational knowledge, and to help them learn and apply new concepts on their own. For me, the most fulfilling aspect of being an educator is helping students overcome the obstacles in their learning.

**Courses I can teach.** My background has prepared me to teach a variety of courses: intro CS courses; intermediate courses on data structures and algorithms, operating systems, and networks; and courses related to security and privacy at any level. Given my research interests, I would enjoy teaching courses related to security and privacy. I am also interested in designing new courses related to security and privacy, particularly interdisciplinary courses.

**Teaching experience and training**

Teaching is hard. It is a skill that matures over one’s lifetime. To improve my teaching, I am always keen to leverage training and learning opportunities. I have attended teaching seminars at both Dartmouth College and the University of Washington. I have also participated in the Future Faculty Teaching Workshop Series, which involved seven two-hour teacher-training sessions, offered by the Dartmouth Center for the Advancement of Learning. In these hands-on workshops, I learned about critical moments in the class, teaching techniques, designing lessons and course syllabus, and the principles of learning and how to apply them. I have incorporated these lessons into my teaching approach.

I have gained teaching experience by teaching an undergraduate course, by giving guest lectures, and by serving as a teaching assistant for both undergraduate and graduate courses. During my postdoc, I had the opportunity to teach the course Data Structures and Algorithms for non-majors in the Allen School at the University of Washington. I received course material from the previous offerings of the course, but I was responsible for all other aspects of the course. The class had 202 students from several non-CS disciplines; it was a difficult class to teach due to its large size and the diversity in students’ CS backgrounds. During my time in teaching this class, I learned a lot about classroom and course management and about teaching students with diverse backgrounds. It also helped me assess my teaching style and develop my teaching approach, which I describe below.

**Teaching philosophy and approach**

At the core of my teaching philosophy is my firm belief that **no concept is too hard to grasp for anyone who wants to learn it**: A willing student simply needs someone willing to explain the concept in a way that the student can understand. This philosophy guides my teaching approach and drives me to find better ways to communicate ideas and concepts to students. Next, I describe my current teaching approach.

**Design class for effective learning.** I design each class with clear learning goals, which I present at the start of the class, and review them at the end of the class. I motivate learning goals with real-world applications or examples familiar to the students. To keep students engaged during class, I structure my class with small learning units with breaks for active-learning exercises. I have used think-pair-share and in-class worksheet activities and have found them effective in my teaching; my students particularly liked the in-class worksheets. I am keen to learn and apply other active-learning techniques, such as reaction to a video and small group discussions.

**Focus on understanding.** To learn effectively, it is important for students to understand a concept than memorize a fact. My teaching emphasizes explaining *why* rather than *what*. For example, when teaching insertions in AVL trees, in addition to identifying specific rotations to do, I also explained insertions using several examples showing why those specific rotations are necessary. I emphasize my focus on understanding in my classes and convey through quizzes and exams by formulating questions that reward deep understanding. Exams and quizzes assess student learning, but they are also teaching opportunities. I try to utilize these opportunities by formulating questions that challenge students to apply concepts in new ways. I found creating such questions challenging but also a rewarding experience.

**Focus on class needs.** I adapt my teaching to the needs of the class by actively seeking feedback from students throughout the term. I seek feedback through in-class worksheets and anonymous online surveys conducted at different times in the term. At the start of the term, I conduct a pre-class survey to ask students about their goals for taking the course, their background relevant to the course, aspects of the course they find most exciting, and what helps them
learn. During the term, I conduct two-to-three surveys to ask students about aspects of the course they find helpful, aspects they find distracting, and things I can change to help them learn better. I used this approach to get feedback in my Data Structures and Algorithms class. From the feedback, I modified class design (e.g., introducing in-class worksheets) and adapted my teaching style (e.g., increasing content and visuals on slides). Seeking ongoing student feedback allows me to improve my teaching immediately for the in-progress course.

Pay attention to individuals. I try to pay attention to the progress of each student in my class and try to offer personalized teaching and support. Students come from varied educational backgrounds and face different challenges: some learn at different speeds, some are shy or anxious about asking help, and some do not have much CS background other than what is taught in classes. Throughout the term, I pay attention to students’ learning (from their in-class worksheets, quizzes, and exams) and proactively help those who are struggling despite trying hard. In my Data Structures and Algorithms class, I sought help from my TAs to identify struggling students, and we identified six students. I enlisted help from five TAs, and we tutored these students one-on-one to address gaps in their knowledge and to build their confidence in solving problems in the course.

Account for diversity. I try to design lessons that cater to all students by considering their backgrounds and pre-existing knowledge. My pre-class survey is instrumental in helping me understand the composition and needs of my class. In mixed-ability classes, I try to provide ample support to students who lack sufficient background by pointing them to resources (e.g., specific chapters or lectures from a prerequisite course) and, when possible, offering extra office hours to help them learn the necessary concepts as quickly as possible. And in office hours, I first ask students questions to understand their current knowledge and use this knowledge as a starting point to answer their questions and to explain concepts. Since different students learn differently, I design my classes using differentiated instruction strategies, which involve providing students with alternate avenues to learn concepts. For example, in my Data Structures and Algorithms class, I explained concepts using different examples and mediums, including Powerpoint animations, videos, and sketching on a tablet; the in-class worksheets also served as an additional learning medium. I also made lecture recordings available to students to review at their own pace. In my future courses, I intend to provide students with alternate learning sources (textual and graphical) for the concepts covered in the class.

Mentoring experience and approach

During my PhD and my postdoc, I advised fifteen students: thirteen undergraduates, one master’s student, and one PhD student. I have worked with these students on both individual and group projects and on short-term (a few months) as well as long-term (12-16 months) projects. Because I use research methods from both systems and HCI research, I have been able to support students on different types of projects (e.g., building systems, conducting field studies, conducting interviews, analyzing datasets using quantitative and qualitative methods). For me, interacting with students and seeing them grow intellectually has been an enriching and rewarding experience. It is always a moment of pride for me when student projects lead to publications, presentations, or open-source artifacts. My students’ projects have led to one publication (co-authored by an undergraduate student), one conference poster (led by the master’s student), two senior theses, and poster presentations in research events.

I have found every student to be different. Some students need a more hands-on approach, some flourish when given free rein, and some need a balanced approach. As an advisor, I adapt my advising style for each student by understanding the student’s strengths, limitations, interests, what motivates them, and how they learn. When advising undergraduate students, I start with a hands-on approach: I work with them to scope their projects and to divide their projects into smaller actionable tasks. When advising graduate students, I use our initial meetings or a small project to assess them and then adapt my advising style to guide them, encourage them, and challenge them to become independent researchers. Overall, my mentoring approach involves helping students choose projects that they find exciting, scoping their projects appropriately, enabling them to take ownership of their projects, and providing the necessary support, guidance, and encouragement.