I value thoughtful, effective, and inclusive teaching. I entered the field of computer science, in part, because of professors who took time to teach hard concepts in an approachable manner. I want to help a new generation learn about computer science. Throughout my PhD, I have had many experiences teaching and advising. For example, I created a DEI course name CS-JEDI for CS PhD students by CS PhD students and received a **best paper award** at SIGCSE for a paper describing the course's creation. I am excited to continue teaching and advising students as a professor.

Teaching Experience

Throughout both my undergraduate and graduate studies, I have furthered my teaching and mentorship abilities through teaching, taking pedagogy courses, developing a course, and advising students.

Wide array of teaching experience, particularly in systems. Through many teaching experiences, I have learned to tailor lectures to different types of classees and strategies to administer larger courses (around 100 students). During my undergraduate degree, I developed and taught several lectures for an introductory online CS course[1]. I tutored **5 different classes** at Harvey Mudd — from introductory CS to computer systems to programming languages. In grad school, I was a teaching assistant for a senior-level undergraduate class on parallel computer architecture and programming and a graduate class on storage systems. I have also been invited to give **6 guest lectures** in various systems classes.

Knowledgeable in CS pedagogy. In addition to hands-on teaching experience, I have learned about the basics of CS pedagogy as part of my graduate studies, which will help me create effective courses. I learned how to develop different components of a course using well-researched techniques to improve student learning: from designing course-wide learning objectives and aligning them to course assessments to how to integrate different types of active learning into classes.

Developing a DEI Course for CS PhD Students. I used these skills to co-lead the development of a PhD-level course, *CS-JEDI: Intro to Justice, Equity, Diversity, and Inclusion in Computer Science* (CMU 15-996)[2]. I worked on creating this course over 2 years, iterating from a remote 3-week pilot to an in-person 6-week course. We made sure to integrate evidence-based design techniques including Universal Design for Learning[3], to lower barriers to learning through student choices, and Culturally Responsive Teaching[4], to recognize the different experiences that students bring to the classroom. We worked with our department's faculty over the 2 years to introduce CS-JEDI as a requirement for new PhD students in CMU's CS department. We also published a paper in SIGCSE, a top CS education conference, and received a **best paper award**.

Mentoring and advising students. I have also frequently mentored and advised students. On my research projects, I mentored **four undergraduate, six masters, and four junior PhD students**. From mentoring these different students, I learned how to size projects for different students based on their interests and experience, starting with small and concrete projects and then gradually giving them larger and more independent projects as their research abilities grow. Two of the undergrads have ended up as co-authors on my published work after they learned how to frame research questions and create experiments to answer them. One of my undergrads, Sophia (Qingyang) Cao, won the SOSP 2024 Student Research Competition, having grown a small analysis project into a project with potential for a top-tier publication.

[1] HarveyMuddX: CS For All: Introduction to Computer Science and Python Programming

[2] Flanigan, Joshi, *McAllister*, Vajiac. *CS-JEDI: Required DEI Education, by CS PhD Students, for CS PhD Students.* SIGCSE 2023.

[3] CAST. 2018. The UDL Guidelines.

[4] New America. 2019. Understanding Culturally Responsive Teaching.

Teaching Methodology

In my teaching, I integrate learning goals into my courses, infuse my lectures with active learning, and deploy inclusive teaching practices.

Establish learning goals at all levels of course design. I want to integrate learning goals into my courses to provide structure, helping both me and my students. They force me to ensure I am prioritizing teaching the material that I want students to learn, whether that be a high-level skill, such as understanding that all system designs have trade-offs, to lower-level skills, such as how to implement basic lock protections for concurrent processes. These goals also allow me to align lectures, homework, and assessments directly to the ideas and concepts that I believe students need to know. For students, explicitly stating the goals gives them a roadmap for what the course is covering and a way to evaluate if they are understanding the content of the course. I used learning goals both in designing CS-JEDI and deciding the contents of my guest lectures.

Integrate active learning during lectures. I also believe in including active learning in classes. Active learning allows students to practice concepts that we cover in class with low-stakes. Students find out that they are understanding the topics while they have the ability to ask questions of the instructor. These exercises also provide valuable feedback to me as an instructor about how well the class is understanding material, allowing me to adjust my pacing and level of explanation to the needs of the class.

I have incorporated various types of active learning. I frequently use think-pair-share activities since they can apply to many types of concepts and give the students some time to think for themselves. I have also used simulations, such as in an online thread synchronization lecture I developed during the CS Pedagogy class. Each student pretended to be a thread that was trying to update a shared counter in Zoom's chat feature. This allowed them to witness the problems that arise from data races first hand.

Create inclusive classrooms. I strongly believe in the need for inclusive learning environments. As a professor, I want to be able to create spaces where all students feel that they are welcome to study computer science generally and systems in particular. Instructor's choices, such as changing an intro to CS website's theme, can significantly increase the sense of belonging for underrepresented students[5]. Thus, I try to be aware of small choices I make in the classroom, such as making sure to use names in slides that represent the diversity of students in the class.

In particular, my research area of computer systems struggles with being hard to approach because of a wide breadth of content where the big picture is easy to lose. Students often feel that the area is too broad, exacerbating imposter syndrome particularly in students already wondering if they belong. One of the ways I have been making systems more approachable with students I mentor is to normalize not knowing everything and how to learn new information. For instance, one of my mentees did not know if part of her model was accurate. I also did not know because of little literature on the topic. We reached out to industry contacts and now we have an ongoing collaboration to explore this question that started from normalizing not knowing everything.

Teaching Interests

From my experience in both teaching and research, I am confident teaching many different systems classes to both undergraduate and graduate students. I am interested in teaching courses on operating systems, storage systems, and distributed systems. I also would like to develop a graduate course on sustainable computing that discusses the environmental impact of computers and how to mitigate this impact from hardware to software. I am also interested in teaching courses on the intersection of CS and inclusion.

[5] Metaxa-Kakavouli et al. *Gender-Inclusive Design: Sense of Belonging and Bias in Web Interfaces.* CHI 2018