Teaching Statement

Jason Lowe-Power
University of Wisconsin-Madison

When the UW Computer Sciences department had an unexpected teaching vacancy, I stepped up to teach a semester of the second-year undergraduate lecture course "Machine Organization and Programming" (http://cs.wisc.edu/~powerjg/cs354-fall15/). I was both excited and a little bit nervous to teach 140 students—with only two weeks to prepare. Through this experience, I discovered my passion for teaching, and I found immense satisfaction from seeing the students learn. While I was preparing to teach the first day of the class, my advisor popped into my office with last minute advice: "When teaching, you should optimize for receiving bandwidth, not sending bandwidth." My goal as a teacher is not to broadcast as much information as possible, but to be student-focused and teach based on the students' feedback.

One of my goals when teaching is to use techniques that increase student feedback such as interactive lectures and encouraging post-lecture discussion. In fact, one of my most effective lectures grew out of students' feedback. Although my research has revolved around hardware caches, my first attempt to teach the basics of CPU caches flopped. After I initially taught the basic concept of caching to my class, I quickly realized due to the post-lecture questions that the students knew no more about caches than when they had walked in 50 minutes earlier. The next lecture, I came prepared. I brought a worksheet with a real application and walked through examples of every step of the caching process with the class.

Teaching by example scales from the lecture hall, where I would walk through examples on the board, down to one-on-one office hours, where I follow a student as she or he works through an example. This pedagogy allows immediate feedback with any number of students, and I can quickly make fine adjustments in how I explain a concept. I encourage active learning, and I found my example-focused lectures were highly effective.

At its core, learning is about assembling a set of tools to apply to future problems. I believe that teaching from basics is the best way to help the students discover these tools. Rather than giving students complete black boxes, I aim to help students understand where these tools originate. For instance, in the computer architecture community we often use software simulators to model complex computer systems. I have worked on providing a document that gives new computer architecture researchers a ground-up foundation using one of the most popular tools, gem5. This work-in-progress book is open source and available online (http://learning.gem5.org). I applied this material to a project-focused graduate computer architecture course. In previous instances, the students spent most of their time learning how to use the tool. However, with the addition of my book, the students focused more of their time on research aims, instead of fighting tools.

Teaching from basics and with feedback-driven methods helps the material resonate with the students from diverse backgrounds. Providing multiple different examples of a concept's application helps when teaching to a diverse audience. My first thought on the best way to teach a concept may not be the most effective, and therefore, I work with the class and individual students to adapt to their learning style.

I have experience mentoring a wide variety of ages, abilities, and backgrounds. I enjoy working one-on-one with junior graduate students. I have a reputation as a good mentor and collaborator. Fellow students seek my opinions, look to me for suggestions on technical challenges, and engage with me to identify and develop interesting research questions. As a mentor, I look forward to creating a rich intellectual and collaborative environment where my students are engaged in each other's projects and seek advice from each other as well as me.

Additionally, I worked to expand my teaching background by teaching computer science to 4th and 5th grade students at local elementary schools (https://sites.google.com/site/uwmadisoncsafterschool/). This was an eye-opening experience for me to see how quickly the students absorbed some of the same concepts I taught college students. Students in this elementary classroom came from a diversity of backgrounds and had diversity in previous computer experience. Therefore, I mostly worked with students one-on-one—watching and listening to help move them toward their goal. I was fortunate to have an opportunity to see the amazing things the students accomplished. Early exposure to computer science through programs like this is important to contribute to the growing diversity in computer science and to fill the future computing needs of our society.