Teaching Statement

Yang Zhou

I greatly enjoy the rewards of teaching and mentoring students. For me, the rewards consist of two parts: (1) the pride and fulfillment when I help students carry out their studies smoothly and when my students grow into independent researchers, and (2) the interesting future research directions inspired or confirmed during teaching and mentoring. Driven by these rewards, I have taught as a teaching assistant for a system course and as a small-group "supervisor" for an algorithm course, and mentored four undergraduates and five junior PhD students in their research. Based on my research background, I am qualified to teach undergraduate courses of computer networks, operating systems, distributed systems, and algorithms and data structures; I would also like to hold three advanced graduate courses:

- *Data center networking:* I will discuss how modern data centers design and build high-performance network fabrics including topology, routing, congestion control, fault tolerance, load balancing, etc.
- *Dataplane operating systems:* I will discuss how the OS evolves to keep up with the fast hardware in data centers, including user-space networking, efficient threading, light-weight isolation, etc.
- *System seminar course:* I will invite a broad set of system researchers from both academia and industry to give talks on various system research topics, and foster potential collaborations with students.

System course teaching: I was the teaching assistant (TA) for a computer system course, the Harvard CS145 Networking at Scale, along with an undergraduate TA. This course features eight P4-switch related projects, three of which are designed and developed by me including detailed guides and skeleton code. I held three one-hour sections covering network programming, background knowledge for projects, and handy tools for developing and debugging. Other duties include holding weekly office hours, answering students' questions on forums, and grading projects. I also had a guest lecture experience at UC Berkeley on far memory techniques in data centers, mainly facing junior graduate students from architecture areas. I started from common and accessible facts like resource utilization and DRAM prices, then explained why data center operators have an interest in far memory, and finally discussed my work in this space.

Algorithm course teaching: I was the small-group supervisor for the Algorithm Design and Analysis course at Peking University as an undergraduate. This role requires supervising around 14 students in small classes, giving recitations, teaching advanced algorithms and data structures, preparing new problem sets and quizzes, and grading, all on a weekly basis. I extensively introduced non-textbook topics related to my undergraduate research of probabilistic data structures. Although time-consuming, being such a supervisor is truly gratifying, especially when students understand my research and try various optimizations as their final course projects. One student (Yicheng Jin) in my small class is now pursuing a computer science PhD at Duke University.

Introductory teaching: I taught non-CS audiences about the Internet from a computer science perspective during the English Language Program at Harvard. It was a slightly difficult yet fun experience especially when I told the audience that Internet data is transmitted in small packets: they were shocked and immediately asked why, and then I gave them detailed yet understandable explanations until they grasped the design philosophy behind it. This experience gave me a good sense of how to teach introductory courses in the future.

Mentoring: I have mentored four undergraduates on system research, and informally mentored five junior PhDs on their research ideas, internship applications, and study experience at Harvard. Among the four undergraduates, one (Zezhou Wang) published an NSDI'23 paper with me and went to University of Washington (UW) as a system PhD; two of them (Xingyu Xiang and Matt Kiley) co-authored an NSDI'24 submission with me, and are about to apply for system PhDs as well as the rest one. Such mentoring brings me enormous pride, e.g., seeing Zezhou gets into the UW PhD program. It inspires my future research—working with Zezhou on eBPF sparks two follow-up projects: one has become the NSDI'24 submission, and another is showing promising results.

My teaching philosophy has been on three aspects:

- *Building safe and inclusive environments.* Students in the same class usually have different prior knowledge; thus it is important to create safe and inclusive environments to make students feel they are welcome to ask both the simplest questions and challenging ones.
- *Focusing on hands-on experiences.* I believe the best way to learn computer systems is through reading, running, debugging, and hacking well-written codebases in a hands-on manner. For future system courses I teach, I would like to incorporate well-written teaching systems, such as the WeensyOS [1], into my agenda to help students gain hands-on experiences.
- *Promoting critical thinking on the pros and cons of techniques.* I would like to encourage students to critically think about new techniques around us, such as the emerging LLMs.

References

[1] Eddie Kohler. Harvard CS61 Systems Programming and Machine Organization (2023): WeensyOS. https://cs61.seas.harvard.edu/site/2023/WeensyOS/.