Teaching Statement

Ayush Mishra

☎ (+41) 0782515146
✉ aymishra@ethz.ch
ayushmishra.net

I've always felt most fulfilled when I'm working with students - in fact, it has been one of the key motivators for me choosing a career in academia. I view it not only as a way to contribute directly to a student's professional development, but also as an opportunity to inspire them to grow as thoughtful, compassionate individuals. Therefore, I have always actively sought opportunities to teach and mentor students during my PhD and Postdoc. These experiences have given me a strong foundation in teaching, mentoring, and evaluating students with a broad range of experience - from incoming freshmen to now PhD students.

Previous Teaching Experience

During my PhD, I taught several undergraduate and graduate computer science courses, earning multiple teaching awards and consistently receiving excellent student evaluations. I was a teaching assistant (TA) for CS1010X: Programming Methodology at NUS for two consecutive years. This summer course was designed for incoming undergraduates transitioning from military service, making it a particularly unique and rewarding teaching experience. I conducted weekly tutorials, guided students through assignments, and provided timely, personalized feedback through our in-house teaching platform, Coursemology. Managing a classroom of students with widely varying technical backgrounds taught me how to balance depth and accessibility. I had to find creative ways to keep the advanced students engaged while ensuring that foundational concepts were clear to everyone. I found that explicitly articulating the thought process behind solving programming problems was especially effective in reinforcing core principles. This emphasis on clarity and conceptual understanding has become a hallmark of my teaching style and is consistently highlighted in my student feedback.

I also served as a TA for CS1231S: Discrete Structures, an undergraduate course introducing logic and proofs to computer science students. I conducted weekly tutorials and guided students through problem sets designed to reinforce foundational reasoning skills. I particularly enjoyed helping students translate intuitive ideas into rigorous arguments and improving their communication skills. Midway through the semester, I introduced a classroom activity, called the *The Wheel of (mis)Fortune*, to encourage participation. Turning the review sessions into a game created a lively and engaging learning environment, motivating students to contribute more actively and articulate their reasoning with confidence.

Beyond undergraduate teaching, I have also been a part of developing and delivering several graduate-level courses. At NUS, I was the TA for CS5229: Advanced Computer Networks, a graduate-level networking course with a strong project component. I helped the main lecturer for this course revamp it entirely from the ground up and prepare the lecture slides, exercises, and other teaching materials. This was an extremely instructive experience in learning what it takes to plan a new graduate-level networking class from the ground up. During my Postdoc at ETH, I have been involved in teaching and organizing a seminar-style course, Seminar in Computer Networks, and teaching Transport and Congestion Control block for Advanced Topics in Communication Networks, a graduate-level networking class. Organizing the seminar was a unique experience - we structured each session to resemble a PC meeting at a networking conference, allowing students to experience the peer-review process and develop critical evaluation skills. I also thoroughly enjoyed overhauling the Transport and Congestion Control block for the graduate-level networking class here at ETH - it was a rewarding experience to try to strike a balance between covering the basics and connecting them with the most recent advancements in this area.

In addition to formal teaching, I have mentored several undergraduate and graduate students for their theses and internships. I take great pride in the growth and achievements of my mentees and continue to maintain lasting professional relationships with many of them. Seeing their continued success and receiving their messages seeking advice remain among the most fulfilling aspects of my academic career.

Teaching Philosophy

Teaching computer science well is uniquely challenging because it blends both science and engineering - disciplines that demand very different approaches to instruction. In my teaching, I work to strike a balance between giving students meaningful hands-on experience and encouraging them to think deeply about the problems a system is designed to solve, along with the assumptions and insights that shape its design. I also try to encourage them to be more conscientious and independent, and remain curious enough to always ask *why* something is the way it is.

Intent-driven Learning. While introducing new concepts and ideas, I like to first introduce the key problem, and the motivation behind solving that problem to the students. I use these to encourage them to come up with their own solution first, before we discuss the new concept. In my opinion, not only does this encourage critical thinking, but it also puts the students in the right head space to appreciate the new idea and remember its key insights better. I try to place special emphasis on this not only to aid better learning, but also to help students identify the *secret sauce* in most good problem statements and their solutions.

Practical Experience. Computer Science is a deeply practical discipline. It not only encourages abstract thinking but also challenges us to turn ideas into tangible systems. In my experience, working hands-on on well-formulated projects can help bridge theory and practice, and better demonstrate to the students where *the rubber meets the road*. For this reason, I like to incorporate project-based learning into my courses, where students translate key networking concepts into working proof-of-concept systems. These projects provide an opportunity for students to engage creatively with the material while gaining familiarity with the tools and frameworks that underpin modern networking systems.

Having an impact beyond the classroom. I also strongly feel that how we *learn* science goes beyond how we just participate in science. Practicing to look for important problems, being methodological and *scientific* in finding solutions, and being comfortable with asking *why* more often are traits my students benefit from outside of their professional lives as well. A classroom is a place where we discuss new ideas all the time - and how we discuss them are great opportunities in teaching students to be kind, empathetic, and be better communicators. A field like systems and networking - where design is all about tradeoffs, and the 'correct' answer is always subjective - is a great advocate for nuanced thinking and contextual reasoning.

Managing Creativity. Teaching science is also a challenge in managing creativity. Again, I feel how we teach new ideas is extremely important in fostering creativity in students. In my experience, junior students often tend to be more creative because their thinking is not yet burdened with *standard practices* and other conventional wisdom. I feel intent-driven learning plays a significant part in maintaining this creativity.

Developing Courses

Given my experience, I feel well-equipped to teach and develop both undergraduate and graduate-level courses. At an undergraduate-level, I would look forward to working on courses that focus on core computing concepts, like Programming Methodology, Operating Systems, Computer Architecture, and Networking. My focus for these courses would be to make them more project driven (I am big fan of the Nand2Tetric course) with clear, tangible objectives. Since these undergraduate courses typically cover several foundational concepts in computer science, I feel it would be especially effective to help students learn these concepts as solutions to practical problems they face in their class projects.

At a graduate-level, I feel the most qualified to design courses covering advanced topics in networking. I have been fortunate enough to help develop and deliver graduate-level networking classes at both ETH and NUS - both of which have been very instructive in teaching me what makes a successful advanced networking class.

Besides this, I am also keen to spearhead efforts in improving tech-literacy, and perhaps developing and offering courses to help older folks keep up with the latest technology, the advent of AI, and the increased digitization of services. As computer science accelerates human progress, we need to make sure that no one gets left behind!