

Teaching Statement

Dinghao Wu

I am passionate about teaching. Being a college teacher is what I have wanted to do with my life ever since I was in high school. I believe that teaching not only shares knowledge and ideas, but also contributes to research in a very positive way. I have participated teaching in various settings from higher education to child programs, from which I have developed my own philosophy.

At Princeton University, I was an Assistant in Instruction (AI) for two introductory level courses, “Computers in Our World” and “Computers and Computing,” both meant for humanities and social sciences students. My responsibilities included holding office hours and grading assignments for classes of 87 and 29 students. With little backgrounds in computer science, mathematics, and engineering, most students needed extensive help on quantitative reasoning and basic computer science concepts, as well as programming and creating home pages. With a college degree in chemical engineering, I especially respect students’ diverse backgrounds and learning ability, and always tried to create a supportive environment. As a result, my office hours often attracted a large number of students. To help students learn efficiently and effectively, I started by asking them a series of questions, guided them through the thinking process step by step, and encouraged them to pursue answers by themselves. That way they not only got the answers, but more importantly, learned how to get the answers.

I also helped teaching the class “Unix & C” in Spring 1997 at Nanjing University. My responsibilities included designing and grading assignments, holding office hours, as well as giving weekly review sessions. Working as a teaching assistant, somewhat as the interface between the teacher and the students, I have come to realize that often there is a gap between students’ understanding and teachers’ expectation. I believe that effective teaching requires a nurturing environment in which students are encouraged to give frank feedback so that the teacher can make responsive adjustments in teaching.

In Summer 2001, I volunteered in the Summer Programming Experience (SPE) program at Princeton University for mentoring an undergraduate student to implement a compression program. I not only gave advice on her programming and helped debugging, but also explained my research to her occasionally. This one-on-one interaction has turned out to be a great experience. It was at that time that I realized the positive effects of teaching on research. The best way to understand a problem thoroughly is to explain it to someone else. Also, the student often had very different views from me, which intrigued me

to give the problem a fresh look and explain it in a way that she could understand. Eventually this exchange of ideas and perspectives not only benefited the student, but myself as well.

I have also had teaching experience in a non-higher-education setting. Driven by my enthusiasm about teaching and my love for the game of Go, I volunteered to teach Go in a local school in the fall of 2002. I taught students, aged 5 through 16, the game rules, local tactics, global strategies, and typical good moves in certain configurations. I also created various intriguing Go problems for the weekly assignments to stimulate students' interests. It was in this very different setting that I recognized the importance of visualization and inspiration in teaching. Students, especially kids, are easy to be distracted, and visual teaching is an effective way to get their attention. Visualization also helps explain things more clearly. Inspired students start their own paths of thinking and discovery. In higher education, if we can inspire students to be interested in computer science, we attract fresh minds into the field, which facilitates the development of the field.

Besides these experiences, I have also attended many training programs in the McGraw Center for Teaching and Learning at Princeton University, including *The Scholar as Teacher*, *Workshop on Teaching Portfolios*, and *Princeton AI Training*. Through these training programs, I have improved my teaching skills in many aspects.

I am interested in teaching introductory computer science courses, as well as advanced undergraduate and graduate level courses and research seminars. My favorite topics include programming languages, compilers, software engineering, program analysis and verification, language-based security, and automated theorem proving. I plan to design seminar courses that would complement the current curriculum for senior undergraduates and graduates. The courses will mainly focus on the latest research in programming languages and software engineering. The material will be covered by research papers, and students will be asked to work on research projects. Such classes will expose students to the frontier of my research area, and will facilitate further collaboration that helps me establish a research group.