

# TEACHING STATEMENT

John Harney (jfh@cs.uga.edu)

I have learned over the past few years that the study of computer science (CS) is somewhat unique to higher learning. It is a relatively new field when compared to traditional subjects, is naturally cross-disciplinary, and is constantly evolving. Thus, the challenge of relaying the understanding of computing to others is formidable. Fortunately, I have had the opportunity to play many different roles in the CS academic setting to help me prepare for a career in CS education. Besides being a student for several years, I have also served as a teaching assistant and lead instructor for various courses. Through these experiences, I have learned that teaching CS is not solely about the material - it is about how to convey the material to the students. I have found that the best teaching practices hinge on three essential abilities: for an educator to (i) develop relationships with students, (ii) challenge students, and (iii) consistently adapt and improve.

A crucial skill that every teacher should possess is the ability to develop strong relationships with their students. Often, I believe this basic skill is taken for granted. When a professor is apathetic, intimidating, rude, or discouraging, students cannot prosper in their academic careers. This is devastating for students who may otherwise have great potential for success. Equally devastating are the adverse effects on the educator, who not only misses an opportunity to contribute to the students' intellectual growth, but fails to attract qualified students to their research programs. The most successful teachers I have encountered as a student (and whom I view as the greatest influences in my teaching experiences) are those with which I am most comfortable. They exhibit patience by realizing students need time to fully understand novel principles. My advisor is a great example of someone who possesses these characteristics. In the first year of my doctoral program, I proposed an idea about improving Web service compositions (the main focus of my dissertation) to him. I had no prior experience in this area, but he was willing to listen and collaborate with me. As a result, we published the idea in a prestigious conference, and have subsequently worked well together the past few years. Coincidentally, I can confidently say that his research program is flourishing and students learn much in his classes.

While developing a rapport with students is important, it is equally essential to make the learning environment as reasonably challenging as possible. Challenging the students, especially CS students, trains them to critically think about problems and pose solutions. In the classroom, this may entail asking students difficult, non-trivial questions and leading discussions on concepts that are more advanced than the pre-determined daily topics. I often touched on topics like algorithm complexity, multithreading, and learning agents in my introductory programming classes and found that the students were genuinely intrigued. Many students, however, have been thoroughly trained throughout their childhood to "make good grades", and will often complain about the difficulties that come with the challenges. I found that I have to be more subtle to overcome this manner of thinking. One way is to assign projects that are more exciting and realistic than traditional abstract applications (Hello World applications are boring!). I also found that I could exploit their penchant for wanting to make a better grade with extra credit opportunities. I remember one former student of mine in particular, whom I recently encountered in a non-classroom setting. He was a business major and was often vocal about his frustrations with the "rigors" of an introductory Java programming course. As his grade was slumping, I gave him an opportunity for extra credit by assigning a project that entailed building a database in MySQL. From this experience, he started to develop an interest in using databases for projects in his other courses and is now employed in a position requiring a lot of database administration, which he thoroughly enjoys.

Finally, educators must continually evolve their teaching styles and strive to improve themselves. The most effective professors I have known often add new elements to each class they teach, and this is something I have tried to emulate. In my first semester as an instructor of record, I lectured nearly every class in a very structured and rigid manner. Being a new, inexperienced teacher, I thought it was a safe, conservative approach that was appropriate for teaching an introductory programming course. While the students performed well, evaluations indicated that they were not inspired and were often bored. When my next teaching opportunity arrived, I lectured far less, coded examples in front of the students, and facilitated open discussions in class. This in general proved to be more successful than the aforementioned lectures, but new challenges emerged. I found that the same students were consistently involved in the discussions while others passively observed. As a result, I slightly altered how discussions were held in my next semester of teaching the same class. I would require each individual student to code an example of the topic of the day in front of class (of course, with help from the student audience and myself). The course evaluations over those three semesters gradually improved, but I still find myself wanting to refine my teaching skills so that I can be a more effective instructor.

I believe that following the basic principles outlined above will enable me to be a successful faculty member in a CS department. I would embrace the opportunity to teach courses related to my current areas of research - namely, artificial intelligence, machine learning, and services-oriented computing - and the chance to potentially collaborate with senior level students who have interests in these areas. I also find teaching introductory courses to be equally satisfying. Here, I can inspire both students who are still unsure about their program of study as well as those non-major students that are just simply interested in adding computing to their skill set. In all, I believe an academic career in computer science will be very rewarding and can think of no other occupation I would rather pursue.