

Statement of Teaching Interests

Jonathan Misurda

Department of Computer Science
University of Pittsburgh
6213 Sennott Square, 210 S. Bouquet St.
Pittsburgh, PA 15260

Voice: (412) 624-8835
Fax: (412) 624-5249
E-mail: jmisurda@cs.pitt.edu
Web: www.cs.pitt.edu/~jmisurda

Philosophy

Computer Science as a discipline has a dichotomy in its focus. Half of the material is theoretical while the other half is practical application. Because this dichotomy can exist within even a single course, one of the most important jobs for any teacher of Computer Science is to bridge this gap. It is much too easy to lose a practically-oriented student while investigating the technical or mathematical background of a topic. Despite the seemingly huge disparity in the types of learning that these two aspects require, neither the theoretical nor the practical exist in a vacuum. They each drive the other, and it is most satisfying when as a teacher I can help students see the mutual influence, and thus gain a fuller understanding than just knowing either aspect alone.

As an instructor, I see my main purpose as being a resource to help solve problems and emphasize the application of the theoretical, while not losing sight of the higher-level concepts. Problem solving, both in the pedagogical sense, and in the sense of fixing the inevitable bugs associated with computer programming, is a skill that can benefit anyone. I see it as imperative that I not only show where problems may arise and how to prevent them, but also to teach students how to solve them on their own. There is no shortage of problems in Computer Science or in real life. Problem solving when applied to why a car will not start is no different than when applied to why a program will not run correctly. There is a logical process to investigate problems, and the students who internalize this process should succeed better in Computer Science and hopefully in life as well. It is my job to make this process evident and to instill its value and utility.

To ground and unite concepts, my lectures often start with a theoretical description of the new topic or problem to solve, followed by guided class participation attempting to construct a solution. With several solutions proposed, we then examine the tradeoffs that may make a certain approach better than another. These new concepts are put into practice through weekly labs, usually done in recitations. Quizzes, tests, and take-home projects review the learning progress made and serve to tie larger concepts together.

One of the main benefits of being a teacher while still a student is that I remember those aspects of teaching that have been most successful for me. My teaching reflects those teachers who have had the most influence upon my own education, and there seems no better way to reach my own students than to emulate their strengths and combine them with my own personal style.

As a teacher, I represent more than an opportunity to get a good grade. I strive to help all students gain a passion for Computer Science and learning in general, one that I myself have and attempt to demonstrate in both my teaching and my outside projects. There is a wealth of knowledge and understanding to be gained both about computers and life, much of which is to be gained outside of school. If I succeed in inspiring my students, they will learn on their own how to be better Computer Scientists.

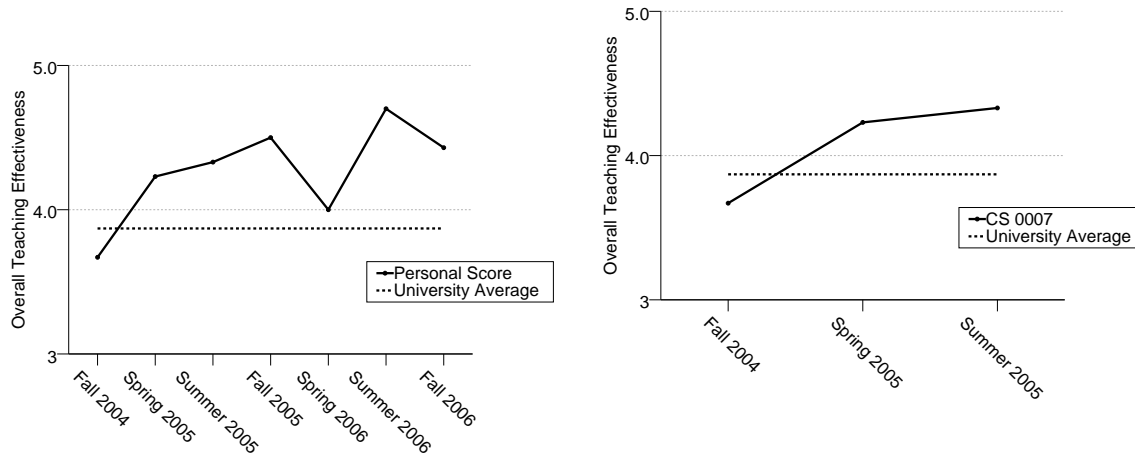


Figure 1: Teaching evaluation scores over time. The left graph is for teaching independently. The right graph illustrates improvement in CS 0007 over several semesters.

Experience

During my experience as a graduate student, I have taught four different courses over eight semesters, and have also been a teaching assistant covering a recitation for two more semesters. Most of my experience as an instructor has been for service courses where I am introducing programming to students with little or no Computer Science background. The left graph in Figure 1 shows my Overall Teaching Effectiveness as anonymously reported during the University administered survey. The overall trend shows that I have improved with experience, and I feel I have learned as much from my students about teaching as I have hopefully taught them about Computer Science. The right graph in Figure 1 demonstrates my improvement across several terms of teaching the same class.

The majority of my teaching experience comes from two classes: CS 0007 – *Introduction to Java Programming* (4 sections) and CS 0132 – *Programming in C and a Guide to UNIX* (2 sections.) CS 0007 is geared towards freshmen who did not take any programming courses in high school. The class begins with a description of computers and how to write algorithms to solve problems. Take-home projects are usually casino games since they are interesting to write and test, and have very definite rules that are easy to express in a programming language. CS 0132 is taken by a variety of students who have had some programming experience before, but need to learn C. They are often adults or graduate students from other departments that need to learn C to do their work and research.

CS 0004 – *Introduction to Computer Programming in Visual Basic* was a very challenging course to teach. The majority of the students were either taking it to fulfill a graduation requirement for a Quantitative Reasoning class, or were in a technology-related but non-programming field such as Medical Information Management. Those students using it to fulfill a graduation requirement were taking the course in place of algebra, and were generally uncomfortable using math. The course greatly challenged these students, and I often provided extra time in office hours for those students who felt they needed additional help. I am proud of all of my students for successfully rising to the challenge of the final project, writing the game of Minesweeper. Programming Minesweeper

required them to demonstrate the same abstract reasoning that they were trying to avoid by not taking algebra. The evaluations from this class represent the dip in Figure 1 in Spring 2006, but I still feel this was a good example of my teaching ability. I look forward to being able to take what I learned from this course and apply it to similar ones in the future to see how I improve.

CS 1590 – *Social Implications of Computing Technology* was a wonderfully different experience to teach. The class consisted of majors who were generally upperclassmen, and the course's focus was on writing and thinking critically about the ethical issues of computers and the internet. Both the students and I had an interesting and insightful time learning and discussing controversial but important issues like net neutrality and copyrights.

Interests

My interest is simply that I want to teach any of the multiple facets of Computer Science. My previous experience has required me to spend considerable time refining my approach to teaching programming, but that is not the full extent of my interest or ability. My research in Software Engineering has allowed me to appreciate the formal design of software and through papers and conference presentations, explain those concepts and my contributions to the field. I would enjoy the chance to bring the foundations of solving problems through good design to students developing maturity in software design.

My graduate classes and additional research has also had a considerable focus on the computer as a machine. By emphasizing the underlying architecture and software interaction via the Operating System or other low-level systems programs, I could bring my experience and insight to students. An extra-curricular interest of mine is mobile computing and I would enjoy bringing my excitement about this field to new and interested minds.

I am also interested in helping undergraduate students plot their college career through classes, internships, and advising. My own undergraduate experience is recent enough that I appreciate the importance of choosing classes that are enriching and stimulating while still meeting the necessary requirements for graduation.

Objectives

My primary goal for being a Computer Science instructor is to educate students for the future, be it as competent professionals or passionate academics. I want to use the experience and education that I have received to help ground the abstract through useful analogies. I also feel a strong responsibility to challenge students to be upstanding and conscientious producers and consumers of technology.

In a practical sense I want to teach how to solve problems with well engineered solutions. This includes a true understanding of the inner workings of the computer. I want to tie concepts across the curriculum into a cohesive picture.

My final goal, but certainly not the least important, is that I want to continue to improve myself and my teaching. I am interested in learning what others have found to be successful methods of teaching Computer Science. I want the chance to reflect on what I have done well or not so well, and take from that and build a better class for the next time.