Teaching Philosophy

Introduction

This brief essay will outline my teaching style and outlook. I believe confidence and competence are closely related, and I do my best to nurture both in my students. I supplement Computer Science theory and concepts with my insights about professional Software Engineering. I have taught mainly students not majoring in Computer Science, and have learned to communicate with them effectively, and I am mindful of social issues surrounding the I.T. workplace and larger technology context.

Competence and Confidence

> "Anyone can cook" – Auguste Gusteau

In the 2007 movie "Ratatouille", fictitious master chef Auguste Gusteau coins the phrase "Anyone can cook" to encourage his T.V. cook show viewers. "Anyone can cook."; anyone can and should take risks success may demand. Anyone can and should take courage and solace in their own competence. Anyone can and should see failure as temporary – bound to an immediate practical problem and nothing else. Similarly, I believe "Anyone can *code*". That is, *anyone* can successfully write software.

Confidence and competence are closely related. Students new to software engineering can feel intimidated. It is important to provide them explicit instructions and walk them through sample problems. Lab and assignment instructions should read more like tutorials than "left to the reader as an exercise". Once students begin to succeed at lab exercises and assignments, their confidence improves to the point where they can more easily imagine – and implement – their own software project. The relevance of the project helps keep their interest and often propels their competence and confidence beyond the scope of the class.

Balance of Theoretical and Practical

One way to help students stay interested in new Computer Science concepts is to relate it to experiences they have outside class. This is especially easy with graduate students or older undergraduate students, since they often have jobs while they complete their education. I share as much of my 20+ year software engineering career as seems appropriate, and with her permission, I also recount stories my wife tells me about working with her I.T. colleagues (she is a researcher, but not an I.T. worker).

The class project is also a way to connect the new Computer Science content students learn to their own interests. Students are encouraged to connect their class project to a hobby of theirs, a future interest, or their daytime job. The project also provides students with an idea of what life is, or could be like as a software engineer.

Teaching Non-Majors

I have never instructed Computer Science majors, only non-majors required to take my courses as part of another major (e.g. Economics or Mathematics) or as electives. I generally assume my audience has no Computer Science background, and that everything should be explained as explicitly and completely as possible. As a result, there is very little room for jargon, technical buzzwords or phrases, sports metaphors, or inside jokes about I.T. This strong requirement has served me well, compelling me to better clarify my thoughts on subject matter and how to explain them in class. Simple, direct communication also makes establishing a rapport with students easier, and discourages the cliques which sometimes form along the lines of technical proficiency or gender.

Social Issues

In addition to connecting Computer Science theory to students interests and work, I ask them to explore the social implications of technology. Depending on the class, we may explore topics such as uneven Internet access (the "Digital Divide"), the influence of large corporations such as Google, Apple and Microsoft on technological advancement, new technologies such as Bitcoin, the Internet of Things, and general application / network security. Each topic is analyzed for practical advantages and disadvantages, but also receives ethical and legal scrutiny.

Conclusion

I have faith in my students; their competence and confidence is my first priority. The core of my pedagogy is class projects, where student interests are incorporated into their learning. I believe technology has a prominent place in modern life, and that students should have a critical view of how technology is applied, reproduced, and disseminated. Everything I've mentioned here, I've implemented in class as an adjunct instructor, and I look forward to working with them again.