

I have gained considerable teaching experience since I started my graduate education at University of Florida (UF) in August 2010. Below I summarize my teaching experience, interests, and philosophy.

Teaching Philosophy

Emphasize understanding rather than rote memorization Through teaching Calculus for Business, I discovered that many undergraduate students have difficulty learning mathematical concepts. A large quantity of effort is wasted trying to memorize, for example, the context of specific algebraic operations as opposed to conceptual understanding of the material. I structured my teaching in order to promote understanding in the following ways: 1) Weekly quizzes and reading assignments emphasized the understanding of mathematical definitions, rather than computational problems. Too often, the foundational definitions are glossed over, and students are unprepared for further development of the material. 2) I organized the class into groups, wherein students were required to discuss intuitive concepts, such as the definition of a limit or differentiable function, with their peers.

Provide practical experience through assignments When designing assignments for Introduction to Bioinformatics¹, I emphasized practical projects that required students to use or implement bioinformatics tools. Furthermore, the students were required to employ these tools using real biological data, which exposed them to preprocessing and other challenges associated with real data. To motivate students to learn abstract concepts, it is imperative to provide insight into the practical uses of these algorithms and data structures. In addition, this experience prepares both undergraduate and graduate students for future employment and advanced studies.

Foster diversity in the classroom Since the 1970s, the recipients of computer science degrees in the United States have been disproportionately white men. Even as other STEM fields, such as biology, award degrees to increasing percentages of women, this is not the case in computer science. One perpetuator of this culture is the *vocal minority* [Franklin, 2013] of experienced students who try to demonstrate their superior knowledge to their peers through excessively advanced questions or complaints about the easiness of the class. These comments can be intimidating to inexperienced students, especially those in underrepresented groups. This situation frequently arose during discussion sections of Software Engineering, and I handled the situation in each case through a private conversation with the perpetrator, during which I asked them to refrain from such comments around their classmates to maintain a supportive atmosphere.

Teaching Experience

As instructor As the sole instructor for Calculus for Business, an introduction to calculus class for business majors at UF, I developed all course materials from scratch, in order to develop an innovative focus on definitions and understanding rather than rote computation. This effort included many *reading assignments and corresponding assessments*, an often overlooked aspect of a mathematical class, designed to promote understanding. Furthermore, I used short Youtube videos and other supplementary, online materials as a resource to illustrate key concepts. In addition, I have served as guest instructor for Introduction to Bioinformatics with Christina Boucher. This involved the design of assignments to promote practical experience, as discussed above.

As teaching assistant All of my teaching assistant (TA) positions have required leading weekly discussion sections and the holding at least three office hours per week. Typically, I was assigned 3 discussion sections of ~ 30 students each per semester, and responsibilities included developing weekly quizzes, working and

¹http://www.christinaboucher.com/teaching/4930/home_assignments.php

critiquing problems for students, and re-presenting difficult material from their lecture sections (typically comprising 200 or more students). In this capacity, I taught Software Engineering, Pre-Calculus, Calculus I, and Calculus II, for a total of 8 semesters. For Software Engineering, I supervised student collaborations on a semester-long project, which typically involved the development of a mobile computing application in the Scrum agile development cycle. This modern, results-oriented approach to software engineering is routinely employed in industry.

Future teaching My experience prepares me for introductory bioinformatics and machine learning courses as well as courses in algorithms or discrete structures courses. In addition, I would be especially interested to develop advanced Data Structures for Big Data and Bioinformatics courses. Many of the principles of teaching mathematics apply in teaching the mathematical analysis and design of algorithms and data structures for computer science.

Student feedback Anonymous feedback from Calculus for Business, in which I was sole instructor:

“I think the suggestion to have quizzes be related to the reading material for the day was a great way to force students to read the material before coming to class, and I found it much easier to grasp the concepts in lecture whenever I had finished the reading at home.”

“Willing to adjust and improve according to the class/curriculum; improved as an instructor as the course progressed (explaining concepts, organizing the course, giving assignments that prepared students).”

“He was really down to earth. Always listened to the students’ concerns, etc. At the same time he kept a rigorous level for the class.”

References

Diana Franklin. A practical guide to gender diversity for computer science faculty. In *Synthesis Lectures on Professionalism and Career Advancement for Scientists and Engineers*, 2013.