My teaching philosophy is really an extension of my personal ideals and life goals. I love being able to help students to succeed in achieving their learning objectives. I truly value and respect the input of all students and feel that I can learn from them as much as I can help them to learn and grow. I feel it is important to not only provide students with materials from the textbook, but also to supplement learning, wherever possible, with real life experiences and additional information from other published sources.

Here are the underlying goals that guide me in my approach to teaching:

• Be a role model for my students, celebrating successes at all levels, and exemplifying acceptance of differences.

• Maintain a positive attitude by living for the moment, recognizing that I cannot change yesterday. Focus on what I can control, rather than what I cannot.

• Teach my students to appreciate and espouse life-long learning. Strive to learn from each opportunity presented.

• Provide service to others through my work as an educator that helps the individuals and community organizations with whom I work to reach their successful goals.

• Accept the responsibility for being a leader in everything that I do.

• Make a difference in people’s lives by supporting the dreams of my students.

• Take time for personal growth and relaxation through daily reading and personal reflection.

During my time at the University of Iowa, I have been the sole instructor responsible for two courses intended for undergraduate students during their second and third years. I chose the text and developed all materials related to the course. These two courses are CS:3210 Programming with Java and CS:3330 Algorithms. In student course evaluations of those courses, given prior to the end of the semester, I consistently achieve ratings around a 5.6 average out of a possible 6 points, which is above the department average of 5.0.

I have taught CS:3210 Programming with Java twice during my time at the University of Iowa. In all courses, I endeavor to go beyond just stating facts about the material. I focus on providing opportunities for students to explore how the material connects to them. For example, the course Programming with Java was created to support providing a solid foundation of Java for informatics students to explore the connection between computational sciences with the arts, the humanities, and the biological, health, information, natural, and social sciences in an interdisciplinary manner with only a light introduction to Computer Science. On specific project worth mentioning from this course is where we work together to create a calculator application. This concrete approach to connections between Java Classes allows students to explore what happens behind the scenes in a single application that many take for granted.

I have also been the sole instructor for CS:3330 Algorithms in three separate semesters. This course covers a wide variety of algorithm design techniques (divide and conquer, dynamic programming, greedy) and analysis techniques (big-0 notation, recurrence); sorting (merge sort, heapsort, and quicksort), searching (B-trees, red-black trees, hashing); basic graph algorithms (depth-first and breadth-first search, minimum spanning trees, shortest paths); and NP-completeness. For this
course, I typically provided a problem to students, or provided them with a list of specifications that needed to be followed in order to solve a problem. We focused on taking familiar examples of various algorithms and making them personal to students. For example, prior to one class we began to discuss pie and ice cream and how several flavors of pie matched better to certain flavors of ice cream. I used this pre-class discussion as a jump off point to discuss the stable marriage problem. Rather than having a list of preferences for both males and females, we constructed a list of best combinations for each pie and ice cream. This small change at the beginning of the semester kicked off an environment that the students commented on after the course finished. I find that these types of small changes increase student investment in the problem for the course because they create parts of the problem rather than just have a method presented to them. The personal ownership component to the material leads students want to be involved in the entire class process, rather than just “check out” because they see something similar.

As a teaching assistant, I almost always was involved with courses (CS I: Fundamentals, CS II: Data Structures, Web Mining, Theory of Computation) where I led a discussion section. These discussion sections were typically smaller components of classes where enrollment totaled over 120 students. A discussion section allowed a single teaching assistant to work weekly with a smaller collection of approximately 20 students to supplement the combined course lecture. During these times, I would run a mixture of classes from lecture to computer science labs. Typically, I would be provided with a topic and would design the materials to present that would cover that topic. For example, when the whole class would be covering runtime of algorithms, I constructed lessons that would explore examples of how to use summations to arrive at a solution.

Additionally, professors felt comfortable asking me to lead lectures in their absences or when they were not comfortable with a topic. I covered numerous lectures for the head of the computer science department Alberto Segre (CS:1210 Computer Science I: Fundamentals). I also covered lectures Brandon Meyers (CS:2230 Computer Science II: Data Structures) and Juan Pablo Hourcade (CS:2520 Human-Computer Interaction). I worked as a team instructor with Sukumar Ghosh covering the specific applications of Java implementations of data structures while he focused on theory during Computer Science II: Data Structures.

I am comfortable teach any course in the standard Computer Science curriculum as well as those courses similar to ones where I have been an instructor or a teaching assistant. Additionally, I am able to lead courses in my PhD area of human computer interaction. Given sufficient time to prepare, I am more than willing to teach any course.

I also worked closely with several undergraduate and high school students as part of my ongoing research endeavors. During this time, I mentored an undergraduate student for two years during which we published four peer-reviewed publications, which included a journal article in *Interactions* and a conference publication at the ACM International Conference on Interaction Design and Children, IDC. Currently, we have submitted an article to the ACM International Conference on Human Factors in Computing Systems, CHI.