

Aspen Olmsted - Teaching Statement

The opportunities to teach students and to develop new pedagogical techniques, and materials are my primary reasons for maintaining an academic career. My academic background, teaching experience, and career as a software architect, developer and manager make me well-prepared to teach many computer science programming courses along with advanced courses on distributed systems, databases, compilers, operating systems, software engineering, networking, data structures, and algorithms.

The most-important thing that we can teach our students is that computer science is interesting, relevant, and fun. A student who is curious and interested in the subject is easy to teach, but, not all students arrive in the classroom in this state of mind. It is my responsibility as a teacher to present the subject in an interesting and engaging manner that shows the elegance and beauty of computer science as well as its applicability to solving concrete and real-world problems and to nurture each student's intrinsic desire to learn.

Over the past twenty-five years, I have enjoyed engaging with many professionals in a teaching capacity. This engagement included programmer management (code reviews, code assistance, internal programming trainings), customer trainings (conference presentations, boot-camp style trainings of database administration, database development, business intelligence development and network management), and customer consulting (small group meetings and knowledge transfer). Over the past three years, I have leveraged my years of industry experience to help engage students in the classroom. My primary technique is to try to illuminate their creativity and entrepreneurial instincts through the use of real life semester-long projects.

I find the traditional weekly programming assignment model used in programming and database classes helps the students to learn the syntax of the new language but fails to teach them the number one skill they need to develop to be successful programmers: problem decomposition. To assist in learning this skill, I introduce a semester-long project that allows students to build a vertical slice of a business solution throughout the semester starting with design (UML class diagrams for programming II and E-R diagrams for database design) followed by several phased presentations that keep the student moving forward and engaged. We use forward engineering to convert our model into code and ultimately produce working applications with GUI and browser based user interfaces. A short presentation is given for each milestone of the project that gives the students the opportunity to learn from others and receive feedback; it also applies the peer motivation which some of the students need.

To further motivate students, I ensure that I demonstrate a love of learning. I do this through a continual pursuit of new knowledge in the form of graduate degrees and professional certifications. I also make it clear, that I want to be at the University by having an open door policy and ensuring I am in my office at least 40 hours a week.

I also find it important to set an expectation for the timeliness of feedback on their submissions. In each class; I give a guaranteed feedback date, and I ensure I meet that goal. I like to use weekly quizzes as a learning tool to help the students measure where they are struggling. If students have difficulty on a quiz, we will revisit the material and add extra questions on the next quiz to reinforce the importance of the material.

In summary, I enjoy computer science and that enthusiasm transfers to students when I can share the correct combination of experience, repetition, inspiration and creativity. I make myself available to students as much as possible to allow that transfer to take place.