One of my greatest passions is to share the things I have learned with others. It is what motivated me to pursue my doctorate and what continues to drive me. At its heart, teaching is simply *learning and learning shared*, without it progress would stop. As leaders in developing the next generation of knowledge, we have the privilege and responsibility to preserve that knowledge by passing it down, both for its esoteric and practical value.

As we engage in a world where almost all aspects of our lives are connected to digital systems, so too does our risk for exploitation. I am excited to teach students how to think in principled ways about their interactions with the digital world, and for the few that pursue careers in computer security, to instill in them its elegance and pragmatic realities. As a systems security researcher with experience developing operating systems, web browsers, and compilers I am positioned to teach courses in systems, security, and compilers including introductory, advanced, and graduate seminar courses, as well as courses on research methodology—similar to a course I created at UIUC.

Experience

While taking an undergraduate course on Signals and Systems (ECE 314 @ UNM), I asked a question during lecture, and the teacher, Professor Andres C. Salazar, precisely connected the edge of my knowledge with the content and took it a step deeper by exploring its broader context. It was exhilarating. This experience has motivated me ever since and established the standard of what I hope to be as a professor. It fueled my undergraduate interactions, where I attribute much of my success to tutoring fellow students. Since then, I have had several formal and informal teaching experiences. First, I served as a teaching assistant for Advanced Operating Systems (CS 523 at UIUC). In CS 523 I created a project where we split into three teams to argue the best OS organization: virtual machine monitors, microkernels, or monolithic kernels? I also facilitated discussion and worked closely with two groups in a shepherding portion of project writings. Second, I created a seminar where students and I explored what it means to become an independent scholar and how to systematically approach the doctorate. In Doctoral Educations Perspectives (DEP), I developed the course from scratch, organizing all readings and establishing a sequence of course exercises and experiences. The seminar culminated in a panel on advising styles from three diverse faculty who provided lively and heated debate. Third, I participated in a course on college teaching (EOL 585) where I learned about knowledge development, and applied it to generate projects and a syllabus for an advanced OS course. Beyond collaboratively building an operating system, the course proposed that students should learn how to read code, and thus included special assignments for peer reviewing other group's systems. Fourth, I had the pleasure of guest lecturing two classes for Introduction to Networks & Security (CIS 331) at the University of Pennsylvania. Last, I have mentored and co-advised several doctoral, masters, and undergraduate students.

Goals

I believe the primary role of the teacher is to facilitate student learning, in the process teaching them how to learn. As a father of four lovely children (and mentor of numerous graduate students), I have learned what I believe to be the central factor in teaching: *more is caught than taught*. Therefore, I put on display everything I want my students to become, both in approach to critical thinking and problem solving. My goal is to connect with students in a way that I can share my passion for learning, and it rub off on them. My hope is to create an environment in which it is impossible to exist without learning.

Beyond that, just as in my research, I care greatly about fundamentally understanding *why* something behaves the way it does, which is always my primary learning objective. *why* transitions a student from rote memorization to understanding. For example, what Professor Salazar did in answering my question was connect the answer with the context and *why* it was like that. *why* also establishes how a course relates to other topics in the field and to student career goals. Computer science is fundamentally about abstractions and translation from one abstraction to another; therefore, these interconnections are vital to deep understanding and continuity throughout the entire degree program. In summary, I believe a full understanding of *why* engenders deeper knowledge structures, long-term learning, and reveals the value of the knowledge being learned, leading to increased passion for course concepts.

Pedagogy

Learning is extremely unique for each person, and therefore requires learner-centered approaches; however, computer science is a discipline that builds from one course to the next, which requires a large amount of factual knowledge. Therefore, in my classes I combine traditional content based instruction with in class assessment techniques (e.g., think-pair-share, ungraded quizzes, etc.), promote active questioning, and create opportunities for group discussion to engage students—students are encouraged and required to critically evaluate ideas on their own and discuss them in class. For example, despite only having two lectures, in CIS 331, I had students devise ways of attacking simple data link layer addressing schemes (*e.g.,* spoofing), or in DEP we started each class session with prompts on readings and journal writing time. The method also serves as a formative evaluation so students and myself are keyed into their comprehension. In addition to creating an active learning environment, assessments inform future instruction as it enables me to directly tie in the new knowledge.

Another characteristic of computer science is that it is team oriented and problem based, and therefore to encourage deep learning and long term retention, I incorporate problem based group projects. I also believe projects are one of the key methods for developing comprehension of *why*. These projects require students to use collaborative technologies, such as Git and Linux, as students will be required to use similar technology in their careers, and convert what would normally be a rote-memorization based course into an active participation where students are creating the very things they learn about, such as an operating system. These projects are concluded with reports that require students to communicate clearly in writing—a trait and skill that should be refined as much as possible.

The last element of learning styles that I pursue is to exemplify expert patterns of thought, skills, and problem solving techniques in front of the class. I believe one of the key ways humans learn is through observation and imitation; therefore, whenever in class and asked a question I always verbalize my thought process, placing particular emphasis on how I dissect problems and attack them. This provides an example to students of an expert's approach, which will enable them to pattern their approaches more easily based on mine. By showing my thought processes, I also demystify the nature of solving what may appear to students as harrowing problems so that they can see the expert going through a step by step process instead of requiring only a single step to find the answer.

Confusion inhibits learning, therefore one of my greatest pursuits in the classroom is to make the course as transparent and clear as possible. Eliminating confusion starts with the initial stages of course planning in the development of the syllabus. I place high effort to coordinate the course and syllabus so students are continually aware of the course goals, learning objectives, my expectations from them, and what they can expect from me. This level of attention not only provides students with a clear map of the course, it also clarifies any points of unfairness as it serves as a de facto contract between myself and the student. The course organization provides students a with tangible demonstration of my commitment. Another key element of clarity is that a teacher must also provide detailed answers to student's questions. When a student asks a question, it is essential for the teacher to understand the particular nuances of the student's misunderstanding, and provide an answer that neither bores nor overwhelms the student, with just the right level of abstraction and application.

Principles

A teacher must be passionate about teaching and more importantly student learning. This passion leads to a greater commitment and enjoyment from both the teacher and student, which aids students in connecting with the teacher. The pith of this passion is that students are important and matter; that they are unequivocally the reason why a course exists. The fact that the teacher treats the course and students with great care and attention, implies students are important, which gives students a sense of being invested in rather than it being just another task. The key is to motivate students by recognizing the challenge they have embarked upon and to *dig in the trenches* with them in their learning journey. This relationship presents a teacher who is approachable and willing to have success and failures with students. It is also directly applicable to mentoring and advising.

A teacher must also be adaptable. Ideally the teacher's grasp of student learning styles and prior knowledge would be perfect. The teacher must be able to recognize when something is failing, and adjust accordingly. When developing a course organization or teaching something new, this problem is bound to arise and the teacher's ability to adapt is key to maintaining student attention and learning capacity. Humility with competence is a great motivator.

Each of these high level principles are integrated in a holistic way enabling an effective learning environment for the student. It is my goal to continually hone the skills described in this statement, and to continually improve upon my understanding of learning.

Mentoring

My passion for teaching deeply extends into the area of mentoring, and my approach is focused on direct engagement and practice of the things I expect my students to learn. I use a pair programming technique to acclimate students to projects and facilitate much quicker productivity. In one instance, I coded for a day then gave control to the student for a day. Working in this way, we had a significant portion of a research prototype (in Qemu) completed in two weeks. Lucian Mogosanu since went on to extend the work to new domains where he recently completed his dissertation. I also care greatly about my students obtaining opportunities for advancement. An undergraduate student, Lei Shi, who I worked with, recently was accepted into the Penn CIS Ph.D. program.

In working with humans, advising must be holistic. As such I spend a significant amount of time learning and understanding each student individually. I typically prefer a direct and engaged advising style, however, with one student I was working extremely hard to direct and focus the research to make it the best it could be. However, I quickly learned that autonomy was essential for this individual, and even though it went against my preferred style I redirected myself to *light-touch-advising* which then led to a successful publication. In another instance I was advising two female first and second year undergraduate students with little experience on coding or operating systems. However, I adapted my techniques to teach them what a buddy allocator is and see them almost complete it in a semester-long independent study.

Overall, I ran into many challenges as a doctoral student, most of which were overcome by some form of tacit rule that can be informally specified and taught. My goal with graduate students is to journey with them as they become independent scholars and give them all the keys I collected along the way. I have formally advised over 15 students and informally through social interactions and being observant several more.