Teaching Statement – Christopher Torng

Being involved with teaching is one of the primary reasons why I am interested in academia. During my time at Cornell University, I have volunteered as an undergraduate teaching assistant, I have volunteered for outreach during my PhD, and I have also twice served as a lead graduate teaching assistant. I have also pushed to enter more of an instructor role and had the opportunity to develop and deliver a few lectures myself for a computer systems programming course. I have been involved in mentorship and teambuilding in both social and academic contexts, including being the music director for a small music group while also leading a small team of graduate students to fabricate two research test chips. These experiences make me excited to continue teaching in the future.

Although I have not taught classes on my own before, there are three key themes of teaching that I have applied in the past and that I plan to continue applying in the future: (1) an emphasis on evaluation of tradeoffs, (2) a focus on modern agile software/hardware development methodologies, and (3) a connection to real-life problems. First, students are often taught to find the correct answers to problems, but solutions to real-world problems are much more likely to exhibit multi-dimensional tradeoffs. My teaching materials will give students the opportunity to evaluate tradeoff spaces with many possibilities and no clear answer (e.g., performance vs. power, programmability vs. efficiency, security, maintainability). Second, a successful career involves not only field-specific knowledge, but also effective development methodologies as well as proficiency with key tools used across the industry. My teaching approach will emphasize the agile software/hardware development paradigm and give students experience with the tools used by small teams to drive complex projects (e.g., version control, continuous integration, project management). Third, I believe that classrooms should strive to connect to real world problems. My teaching approach emphasizes connecting class projects to important emerging fields (e.g., machine learning, the Internet of Things), where students can feel the impact and get excited about their work.

The rest of this statement will first describe how these three key themes played a part in my experience in ECE 2400 Computer Systems Programming, ECE 4750 Computer Architecture, and the CURIE Academy outreach program for high school girls exploring an Internet of Things. I will then discuss my teaching interests.

ECE 2400 Computer Systems Programming

`https://www.csl.cornell.edu/courses/ece2400/2017`

ECE 2400 Computer Systems Programming is a new sophomore-level class at Cornell designed to introduce students to C/C++ programming from a hardware systems perspective. The class teaches data structures and object-oriented programming while keeping in mind the underlying hardware (e.g., memory layout, assembly representation, threads and processes). As the lead graduate teaching assistant for a new class, I had the unique opportunity to co-develop course content and influence the trajectory of the course in collaboration with Professor Christopher Batten. I have always wanted more opportunities to develop my own teaching style, and in this course I prepared my own lecture materials and taught three lectures on hash tables and Unix processes.

I designed the programming assignments around my three teaching themes. First, students implemented a baseline and an alternative approach to the implementation of a data structure or algorithm, and then students evaluated the tradeoffs. For example, students implemented their own data structures (i.e., vectors, lists) and simple algorithms (e.g., variants of sorting) before qualitatively and quantitatively arguing for the most suitable data structures and algorithms (correct answer, "it depends"). The assignments also leveraged modern agile software programming practices (e.g., git version control, GitHub, TravisCI for continuous integration, code coverage). Finally, I connected students to real world problems by designing a machine learning programming assignment based on handwriting recognition systems, in which students evaluated various algorithms for accuracy and performance on MNIST digits (e.g., simple ones classifier, brute-force k-nearest neighbors classifier, a k-d tree-based classifier).

ECE 4750 Computer Architecture

`https://www.csl.cornell.edu/courses/ece4750/2016f`

ECE 4750 Computer Architecture is a senior-level class that expands on the sophomore-level digital logic class. Students learn about basic processors (e.g., simple pipelines), basic memories (e.g., direct-mapped caches), basic networks, advanced processors (e.g., out-of-order pipelines), advanced memories, and other advanced topics. I had extensive experience with this course because I had already taken this course as an undergrad, I had already been an undergraduate teaching assistant for it, and finally I had become the lead graduate teaching assistant under Professor Christopher Batten.

I enhanced pre-existing lab assignments around my three teaching themes. I designed the labs to include qualitative and quantitative evaluation to compare the baseline hardware system with an alternative that was typically more advanced with certain tradeoffs. For example, students compared basic pipelines with interlocking-based hazard avoidance to more aggressive designs with bypassing, before arguing *quantitatively* for performance via simulation as well as *qualitatively* for area and energy given the additional hardware complexity. I again placed great emphasis on agile software/hardware programming practices, with students developing under git-based version control, GitHub, and continuous integration. We also used a novel Python-based productive hardware development framework to emphasize productive hardware design and verification. The final lab assignment was designed to connect to the real world, with students building a multicore system with four cores connected with a ring network — a system reminiscent of the processors likely driving the smartphones in their pockets.

CURIE Academy

`http://www.csl.cornell.edu/curie2014`

The CURIE Academy is a one-week summer residential program organized by the Cornell Diversity Programs in Engineering for high school girls who excel in math and science, enjoy solving problems, and want to learn more about careers in engineering. Under Professor Christopher Batten, I constructed a series of design experiences that spanned across building a simple calculator based on logic gates, to assembling small Arduino-based motor-wheeled robots that navigate a maze to find a target, to designing edge devices for the Internet of Things targeting concerns in healthcare, disaster avoidance, smart homes, and other contexts.

My three teaching themes were apparent despite the short program length. While no report was required, the scholars discussed extensions to the physical design of the robots to enhance their path-finding, but then evaluated the corresponding tradeoffs in software for the additional programming complexity. Students developed coding skills with microcontrollers in an Arduino-based development framework and learned to iteratively test and debug using an incremental development approach. Finally, the students explored an exciting emerging computing paradigm based on the Internet of Things. The final lab experience I designed had each of the scholars program their devices to communicate with the cloud in order to control smart doors, monitor heart rate, detect shaking motions (early disaster warning), and track wildlife.

Teaching Interests

The classes I am well-suited for teaching include introductory programming, digital logic and computer organization, embedded systems, digital systems design with microcontrollers, digital VLSI design, computer architecture, and parallel programming. I plan to apply my three teaching themes across any future classes I develop.

I am also interested in developing a new ASIC-oriented class that allows students to fabricate chips with hardware accelerators targeting important fields (e.g., machine learning, IoT). I value giving students an opportunity to design a finished product that they can hold in their hand (if possible) and add to their resumes. The class would also give students experience working with the growing open-source software/hardware ecosystem that is generating excitement in both industry and academia today. The chips could be fabricated through an educational program with a multi-project wafer service like MOSIS. Cornell University did not have a course of this type, but I feel it would have made a great addition to the department, and I am interested in developing it myself.

Being involved with teaching is one of the primary reasons why I am interested in academia, and I am excited to continue teaching and interacting with students at all levels of education in the future.