

Title: Development of New Course: CS0300 Fundamentals of Computer Systems

Sponsor: Malte Schwarzkopf

Abstract: As a capstone I worked with Professor Schwarzkopf along with a team of TAs to conceive, design, and create a new intermediate course focused on teaching the relevant, fundamental material for computer systems. I was responsible for building a grading infrastructure for the class, with the goals of integrating git and github classrooms, as well as a strong emphasis on auto-grading and immediate feedback for all student submissions. This involved setting up repositories for each project, scripts to synchronize and update the website and grading server automatically. Additionally, I set up a shadow server that allowed us to test the deployment and submission of assignments as they were developed throughout the semester, without interfering with the student-facing grading server.

Toward the end of the semester, I worked with Lisa Phinisee to develop the fourth project for the class focusing on concurrency and threading. For the project, students first had to implement a thread-safe queue class, which they then had to use while they implemented key functions in a banking-style server used to process client requests, like payments, deposits, withdrawals, and charges to other clients. The most challenging part of making this project was making sure the solution code was 100% thread-safe and memory-safe, while using as few C++ specific constructs possible (instead mimicking C, to help with the steep learning curve).

Lastly, I built a generic autograder that was used for all labs, project 4, and project 5. The hope is that next year projects 1, 2, and 3 will adopt the autograder, as it allows for very simple test creation both by TAs and by students, and formats the output nicely (I spent so much time wrestling bash output colors). I wrote it in bash because dealing with signals and processes is less painful than python.

