Samuel Wolfson

206.601.5615 / wolfson@cs.washington.edu
samwolfson.com / github.com/terabyte128

Education

University of Washington, Seattle WA **2015-Present**

- Masters of Science, Computer Science (4.0 GPA) Expected June 2020
  - Selected Coursework: Security, Embedded Systems, Computer Architecture
- Bachelor of Science, Computer Science (Manga Cum Laude, 3.9 GPA) March 2019
  - Selected Coursework: Operating Systems, Compilers, Machine Learning

---

Skills

Languages: C, Golang, HTML/CSS, Java, JavaScript, LaTeX, Python, Ruby.
Frameworks: Django, ExpressJS, gRPC, Ruby on Rails, Sequelize, JavaParser.
Environments: Docker, Git, SQL (MySQL, PostgreSQL), UNIX/Linux command line, Spark.
Miscellaneous: University teaching, course development.

Projects & Activities

Processing to P5.js Compiler **Summer 2018**

- Modified a Java parser and AST builder library to build an AST for the Processing programming language (processing.org) and output code in the the JavaScript version of the language, P5.js (p5js.org).
- This compiler was used by 40 students in an introductory programming class (Computer Science Principles) so they can run their Processing code on the web without manually converting it.

Work Experience

UW CSE: Predoctoral Instructor (Hardware/Software Interface) **June-August 2019**

- Prepared and delivered three lectures a week for a quarter-long course.
- Managed a staff of three TAs and a class of 50 students.
- Led weekly meetings to discuss goals, manage grading load, and plan for the week.
- Developed new grading infrastructure for programming assignments using Docker images on the Gradescope platform.
- Wrote, and developed grading rubrics for, midterm and final exams.

Arista Networks: Software Engineering Intern **June-September 2018**

- Implemented code to automatically power off servers in our test environment when they are not actively in use, leading to a 9% decrease in overall energy usage by test servers.
- Designed a strategy to move management code for test servers out of local user workspaces and into containerized microservices, using Docker to run the services and gRPC to facilitate communication between the clients and the services.
  - This design allows the underlying implementation of the service to change without breaking outdated workspaces.
  - Wrote an implementation of a service (in Golang) using my design that controls the power state of our test servers.

UW CSE: Teaching Assistant (Hardware/Software Interface, Computer Science Principles, Compilers) **2017-2019**

- Led sections, labs, and office hours, where I answered students' questions and explained new & novel concepts.
- Helped to develop new course materials, worksheets and infrastructure; developed and presented lectures.

Progressive Tech: Computer Repair Technician & Internal Support Specialist **2009-2019**

- Developed scripts for network booting of hardware diagnostic tools, allowing technicians to diagnose client computers without using flash drives (which can easily become infected by malware and must be manually updated with new tools).
- Created a website for streamlining and standardizing repair quotes, improving consistency of documentation and pricing for quotes from different technicians.
- Prototyped a new order and customer management system, as a proof of concept for replacement of the existing system.