Daniel Liu

danlliu@umich.edu (734) 585-4865 linkedin.com/in/danlliu danlliu.github.io github.com/danlliu

Education

University of Michigan – Ann Arbor, MI Anticipated Graduation: **May 2024**

Bachelor of Science in Computer Science and Chemistry: College of Literature, Science, and the Arts **GPA: 3.9 / 4.0**
Bachelor of Business Administration: Ross School of Business

Relevant Coursework

Intro. to Operating Systems, Compiler Construction, Intro. to Computer Security, Mobile App Development, Web Systems, User Interface Development, Quantum Computing

Awards

2022 Barry Goldwater Scholarship

Work Experience

Apple Jun. 2023 – Aug. 2023

Software Engineering Intern: WebKit, JavaScriptCore

- Developed a new WebAssembly interpreter for Apple's JavaScriptCore framework, improving startup time and memory usage by 7% compared to existing WebAssembly infrastructure
- Shipped over 14,000 lines of C++ and handwritten assembly to implement the core WebAssembly specification, giving the new interpreter similar runtime performance to existing infrastructure

Amazon AWS: Lab126 May 2022 – Jul. 2022

Software Development Engineer Intern

- Collaborated with full time SDEs and other interns to integrate a front-end Android app developed in Kotlin with existing APIs and new ML models to provide users with a wide range of responsive and intelligent functionality
- Developed and updated user-facing content to improve user experience and incorporate new technical features
- Added quality assurance and user engagement code to provide engineers with meaningful data about what features of the app users engaged most with

University of Michigan College of Engineering Sep. 2020 – present

Instructional Aide: EECS 280, EECS 281, EECS 370, EECS 388

- Served on the teaching staff of three main-track (including Data Structures and Algorithms and Computer Organization) and one upper-level (Intro Computer Security) computer science course, teaching ~5000 students across seven semesters
- Helped students debug coding projects involving C++, C, Python, x86, and Go; understand lecture material through regularly scheduled office hours and lab sections; and build confidence in their coding abilities
- Developed lab assignments, coding projects, course notes, and exam questions to reinforce and measure student understanding of course material
- Collaborated with course staff to ensure a seamless student experience, leading effort on key tasks such as exam creation, grading, and new project development

Personal Projects

dvim | C++, `ncurses`, Makefiles

- Developed a terminal text editor inspired by vim in the course of roughly two weeks, utilizing C++17, the `ncurses` library for terminal output, and GitHub Actions and pull requests for CI/CD
- Designed a modular code structure to improve development efficiency by maintaining effective abstractions for components such as the display and the editor state

cppserver | C++, `absl`, CMake

- Created a Flask-like webserver in C++17, utilizing the C socket API to accept requests, a multithreaded server to handle incoming requests, and a custom template engine to handle server-side dynamic content
- Utilized Google's `absl` library, the `std::regex` library, the `std::filesystem` library, and C++'s threading functionality to implement a variety of important algorithms, such as a thread-based request handler with limited concurrency, endpoint registration through first-class function objects, and abstract syntax tree parsing for the template engine

Skills

Programming Languages:	C++, C, x86, ARM64, JavaScript, Python, Rust, Kotlin, Swift, WebAssembly, Java, Go, SQL
IDEs and Editors:	Visual Studio Code, Neovim, Vim, IntelliJ Suite, Xcode
Software Tools:	git, zsh, bash, Docker, gdb, lldb, Makefiles