Sean Yan

Computer Science Major

linkedin.com/in/dhsyan dhsyan@student.ubc.ca +1-(604)-977-6799 DHSYan

Skills

Languages: Javascript/Typescript, C/C++, Java, Python, Rust, Bash, Scheme, Assembly, HTML/CSS, LaTeX
Tools: NodeJS, Jest, Docker, ExpressJS, MongoDB, Pandas, Git/Github (Action), Flask, Nix, Linux.

Experiences

Mathematics & English Tutor Taichung, Taiwan 05/2023 - 07/2023
Meiko School

- Tutored students from years 1 - 12 in Mathematics and English.
- Improved students' school Mathematics test scores from 46% to 96%
- Helped students gain confidence in the topics of mathematics through lively teaching.
- Written quality reports to help parents better understand students' performance in sessions and provide comments on ways to continuously self-improve in the field of math.

Projects

Job Tracker API

- Designed and developed a RESTful Job Tracker API using Node.js, Express.js, and MongoDB, enabling users to efficiently track job applications with CRUD operations, notes, and status management.
- Implemented robust data modeling and modular architecture with Mongoose ODM, improving code maintainability and reducing onboarding time for new developers by 40%.
- Containerized the entire application with Docker and Docker Compose, simulating a production-grade environment and enhancing deployment reliability across different systems.
- Achieved 95%+ unit test coverage using Jest, ensuring stable CI/CD integration and significantly reducing regression bugs during feature updates.

PRIMPL Assembler, SIMPL Compiler

- Built a compiler and assembler for custom imperative and assembly languages, simulating a full toolchain.
- Gained strong proficiency in manual memory management, low-level architecture, and algorithms.
- Applied deep understanding of stack organization, including frames, pointers, and heap allocation.

Nix Config; Infrastructure as Code (No fear in reinstalling your OS!)

- Developed declarative, reproducible Nix configurations to ensure identical OS setups across systems.
- Implemented infrastructure as code to automate system rebuilds, eliminating manual provisioning.
- Acquired deep understanding of Linux systems by daily driving NixOS, and increased efficiency by 100%

.dotfiles - Personalized Software Developement Workflow

- Engineered declarative and reproducible NixOS configurations, guaranteeing consistent system environments across multiple machines.
- Automated full system provisioning using Infrastructure as Code principles, enabling one-command rebuilds and eliminating manual setup.
- Daily drove NixOS as a primary OS, developing expert-level knowledge of Linux internals and improving personal development workflow efficiency by 10x.

Terminal Chat App (built 100% in C)

- Built a fully multi-threaded Terminal Chat Application in C using TCP/UDP sockets and POSIX threads, enabling concurrent handling of multiple users and real-time communication.
- Designed thread-safe mechanisms to support features such as private messaging, broadcast messaging, and peer-to-peer media transfer without blocking.
- Mastered low-level network and concurrency programming by implementing per-client thread management, synchronization primitives, and custom communication protocols.

Education

University of British Columbia 09/2023 - Present

Program: Bachelor of Science
Major: Computer Science
GPA: 4.33/4.33
Standing: Dean's Scholar