Vladimir Trifonov

vova.trifonov@hotmail.com | LinkedIn | GitHub | Redmond, WA, USA

Education

University of Washington — Seattle, Washington

B.S. in Computer Science; ***GPA: 3.82/4.00*** — *Sep 2023 – Aug 2025*

Coursework: Machine Learning, Quantum Computation, Compiler Construction, Operating Systems, Computer Communication Networks, Data Structures and Parallelism, Computer Security, Systems Programming, Programming Languages, Hardware Software Interface, Data Management, Software Design and Implementation

Skills

Languages: Go, Java, C, C++, JavaScript, TypeScript, Ocaml, x86_64, SQL

Technologies: Linux, Docker, AWS, React.js, Git

Methodologies: OOP, Functional Programming, Procedural Programming

Experience

Big Dawg App, Husky Coding Project — Seattle, WA

Software Engineer — *Sep 2024 – Present, Part-Time*

- Designing and implementing mobile workout logging app
- Working on database and backend
- Cross-platform, written in typescript using Expo (React Native Framework)
- *GitHub*

Paladin Cloud — Bellevue, WA

Software Engineering Intern — *Dec 2024 – Feb 2025, Part-Time*

- Working as backend engineering intern for startup focusing on Saas security monitoring for cloud deoployments
- Refactoring Go code in AWS lambda functions for transition to v2 of product
- Open source work (@vovapaladin and @Vladimirtrif): *GitHub*

Team 949z, Vex VRC — Sammamish, WA

Software and Robotics Engineer — *Oct 2021 – May 2023*

- Built robot and programmed it in C++ for each competitive Vex VRC season
- Programmed autonomous and manual control modes
- Placed top 40 in the Vex Worlds Championship 2022, Semifinals at State 2023
- 21-22 Season: *GitHub* | 22-23 Season: *GitHub*

Projects

MiniJava x86 Compiler | *GitHub*

- Implemented a Minijava (subset of Java) to x86_64 compiler
- Features static type checking and implementation of object oriented programming in x86 with polymorphism and method overriding
- Written in Java with CUP and JFlex

Trefoil Programming Language | *GitHub*

- Implemented a functional, LISP-like, dynamically typed, programming language that is interpreted in Ocaml
- Features first class functions, function closures, parrtially applied functions (currying), and pattern matching

AI Pneumonia Diagnosis | *Colab*

- Trained an AI Penumonia diagnosis model in Google Colab with Python
- This neural network project was made for the team project for the Inspirit AI Scholars Program

Dungeon Raider | *GitHub*

- Developed a side-scrolling browser game from scratch written in vanilla javascript and html
- Created for FBLA Computer Game and Simulation event. Presented at state level in Washinton (WAFBLA)