

# Yiyuan Li

Singapore, River Vally | yy@eyuan.me | 895 234 47 | eyuan.me | linkedin.com/in/yyuanl  
github.com/yuann3

## Summary

Motivated IT undergraduate from The University of Newcastle with hands-on experience in backend development, system design, and algorithm optimization. Skilled in building scalable solutions using Rust, Python, Golang, and C, with a focus on performance and efficiency. Proven through projects like an AI-powered learning assistant, a multithreaded HTTP server, and a lightweight CLI tool in Go. Eager to contribute technical expertise to innovative backend teams.

## Education

<b>The University of Newcastle</b> , Bachelor in Information Technology	Jan 2024 – Sep 2025
<ul style="list-style-type: none"><li>• <b>High Distinction:</b> Object-Oriented Programming</li><li>• <b>Distinction:</b> Data Structures &amp; Algorithms, Advanced Database</li></ul>	
<b>Singapore University of Technology and Design (SUTD), École 42 Program</b> , Completed intensive project-based curriculum; proficient in C, Linux, and Git	Feb 2024 – Oct 2024
<b>PSB Academy, Singapore</b> , Diploma in InfoComm Technology	Jan 2023 – Nov 2023

## Experience

<b>Peer Assisted Study Sessions (PASS) Leader</b> , The University of Newcastle – Singapore	Feb 2024 – Present
<ul style="list-style-type: none"><li>• Mentored peers in Data Structures, OOP, and Web Technologies, designing practice problems and code snippets to reinforce concepts.</li><li>• Delivered technical mini-lectures, improving students' understanding of complex algorithms and coding practices.</li></ul>	
<b>Cadet, Pisciner</b> , Singapore University of Technology and Design (SUTD), École 42 Programme – Singapore	Sep 2024 – March 2024
<ul style="list-style-type: none"><li>• Mastered C programming, Linux, and Shell scripting through 16 projects (12 individual, 4 group) in 26 days.</li><li>• Enhanced collaboration and version control skills using Git in a team of 150.</li></ul>	

## Projects

<b>Pew: Lightweight CLI for Code Dumping</b>	Personal Project
<ul style="list-style-type: none"><li>• Built a Golang CLI tool to consolidate source code or directories into a single Markdown file for LLM workflows.</li><li>• Implemented file parsing, Gitignore-style pattern matching, and syntax-highlighted output with tree-style directory visualization.</li><li>• Tools Used: Golang, CLI Development, File I/O</li></ul>	
<b>Hiraku: AI-Powered Smart Learning Companion</b>	The University of Newcastle
<ul style="list-style-type: none"><li>• Led backend development of an AI assistant using Python, Flask, and SQLite, integrating Retrieval-Augmented Generation (RAG) for document processing (PDF, TXT, etc.).</li><li>• Built a secure REST API with JWT authentication and optimized vector storage using ChromaDB.</li><li>• Tool Used: JavaScript/TypeScript (React 19, Next.js), Tailwind CSS, Python (Flask REST API, JWT), Database (SQLite)</li></ul>	
<b>Rust HTTP Server</b>	Personal Project
<ul style="list-style-type: none"><li>• Developed a multithreaded HTTP/1.1 server in Rust supporting GET/POST, file uploads, and gzip compression.</li><li>• Added a User-Agent echo endpoint for debugging and optimized response efficiency.</li></ul>	

- Tools Used: Rust, HTTP/1.1, Multithreading

### Ray Tracer in Rust

Personal Project

- A high-performance ray tracer was developed from scratch in Rust. It features robust 3D vector mathematics, comprehensive operator overloading, and thorough unit testing. An efficient camera system was designed, incorporating viewport calculations and ray generation capabilities. Rust's type system was utilized to ensure strong compile-time guarantees and memory safety
- Tools Used: Rust, Linear Algebra, Computer Graphics, PPM Image Format

### Ylib: C Standard Library Rewrite

École 42

- Ylib is a modern rewrite of the C Standard Library that aims to enhance performance and maintainability while adhering to the ANSI C standard. It includes core C library components such as string manipulation, memory management, I/O operations, and math functions.
- Tools Used: C, Makefile, GCC, GDB

### Push Swap: Stack Sorting Algorithm

École 42

- Optimized integer stack sorting with minimal operations using C, achieving  $\mathcal{O}(d(n + k))$  efficiency
- Tools Used: C, Makefile, GCC, GDB

## Skills & Technologies

---

### Technical Skills

**Programming Languages:** Golang, TypeScript, Rust, C++/C, Python, Java, SQL, Kotlin, SQL, , JavaScript, C#

**Database & Storage:** SQLite, MSSQL, MongoDB, Vector Databases (ChromaDB)

**Frameworks:** Raylib, React.js, Node.js, MAUI

**Operating Systems:** Unix, Linux(Debian, Arch, Ubuntu), Windows, MacOS

**Tools:** Git, Docker, Linux (Debian, Ubuntu), GDB, Neovim, Makefiles

**Concepts:** Distributed Systems, Algorithm Optimization, Data Structures (Trees, Graphs, Hash Tables),

### Soft Skills

Team Collaboration, Problem solving, Mentorship, Adaptability, Communication, Leadership, Critical thinking, Time management.