Alexander Paolini

+1 (561) 617-6922 | alexander@paolini.dev | github.com/alexanderpaolini | linkedin.com/in/alexanderpaolini | paolini.dev

Education

University of Central Florida, Burnett Honors College — Aug. 2024 — May 2028
B.S. in Computer Science | Minor in Mathematics; Computer Engineering | 3.91 GPA — *Orlando, FL*

- Computer Science Accelerated BS to MS (expected 2029)
- Associations: Knight Hacks, Honors Congress, KnightRiders, UCF Weightlifting, Programming Team
- Relevant Coursework: Intro to C, Discrete Structures, Object Oriented Programming, Computer Science 2

Experience

Ignite Software Engineering Intern — May 2026 — Aug. 2026
NVIDIA — *Santa Clara, CA*

- Incoming Summer 2026

Software Engineering Intern — May 2024 — Aug. 2024
Better World Analytics — *Melbourne, FL*

- Developed **data processing scripts** using **Python** and **pandas** to analyze **call detail records** and **timing advance** tables.
- Utilized **Kepler.gl** to analyze and visualize potential criminal paths, providing insights to aid defense lawyers in preventing wrongful convictions.
- Collaborated with a team through weekly standup meetings, maintaining effective communication via email with coworkers and supervisors.

Volunteer Developer — 2021 — 2023
JPBBots — *Remote*

- Developed many different Discord bot applications including **Censor Bot**, a bot that automatically deletes inappropriate or disallowed words/phrases.
- Led the development of Censor Bot's filter, ensuring comprehensive coverage against inappropriate language, including handling lookalike characters and diacritics.
- Utilized **Git** and **Docker** with **TypeScript**, contributing to a project utilized by over **100,000 communities** and **millions of users**.

Projects

Roopl - Programming Language | *rust*, *assembly*, *three-address code*, *codegen*

- Developed a handwritten compiler for an object oriented programming language compiling to native executables using **x86 Assembly**.
- Designed a **DFS** based module resolution system enabling dynamic imports in roopl source code files.
- Lowered high level **Abstract Syntax Tree** into custom three-address IR before emitting x86 assembly.
- Practiced test-driven development with comprehensive unit tests covering lexing, parsing, IR generation, and codegen.

ibssbi - Bytecode Interpreter | *C*, *Binary*, *Bitwise Operations*, *Register-Based VM*

- Designed and implemented a custom register-based bytecode interpreter in **C** as a foundation for a future compiler.
- Supports execution of **37 OpCodes**, including arithmetic operations, control flow, stack management, and system calls, with room for **64 total instructions**.
- Implements a **64-register** architecture with a dedicated stack and jump/call instructions for execution flow, using a compact 32-bit instruction format.
- Optimized **instruction decoding** with **bitwise operations** and **C Macros** for performance and readability.

DAVe Card - Full-Stack Web Application | *Next.js*, *React*, *Typescript*, *Tailwind*, *tRPC*, *OAuth2*, *CardDAV*

- Built a lightweight contact server and social contacts manager enabling a single canonical contact card to be edited in-app and synced to friends' devices via the internet contacts standard.
- Implemented a minimal CardDAV server – following **RFC 6352** and supporting user authentication contact syncing on **all modern contact applications**, including Apple Contacts.
- Supported login via **OAuth2** with Discord, Google, and Github and a custom automatically syncing **friend-request system**

Honors and Awards

- **1st place** at Lockheed Martin Code Quest 2024
- **12th place** at the 38th Annual UCF High School Programming Tournament
- UCF President's List | Fall 2024, Spring 2025, Fall 2025
- UCF Provost Scholarship

Skills

- **Programming Languages**: Python, C++, C, Java, JavaScript, TypeScript
- **Libraries/Frameworks**: numpy, pandas, Express.js, Next.JS, React
- **Tools/Platforms**: Windows, Linux, MacOs, Git, GitHub, Raspberry Pi, Node.js, VSCode, kepler.gl, LaTeX, Typst