

Meredith Hu

Contact: (917)-689-1177 – mmh264@cornell.edu

GitHub: github.com/meredithmhu

LinkedIn: linkedin.com/in/meredithhu

Software engineer with strong systems fundamentals and tooling experience.

Education

Cornell University, Ithaca, NY

May 2024

Bachelor of Arts Double Major: Computer Science and Classics

Relevant Courses: Data Structures, Algorithms, Computer System Organization, Functional Programming, Formal Verification, Programming Language Theory, OS, AI, Linear Algebra, Probability, Discrete Math

Work Experience

Capra CS Lab, Ithaca, NY – *Undergraduate Research Assistant*

May 2022 - May 2024

- Modeled the N-Queens problem as a constraint satisfaction system using Python and the Z3 SMT solver, programmatically generating constraints for arbitrary board sizes and verifying solution correctness.
- Built a Python CLI tool to automate benchmark execution, compilation, and runtime measurement, replacing manual workflows and reducing sources of human error.
- Iteratively extended the tool with features such as batch execution, statistical analysis (mean, standard deviation), and CSV export, using debugging and user feedback to refine functionality.
- Debugged path resolution, subprocess execution, and environment issues while integrating with an external compiler toolchain, improving reliability of experimental results.

Competitive Edge Tutoring, Wyzant – *CS Tutor & Test Prep Tutor*

Sept 2025 – Present

- Support college students virtually in algorithms, circuits, and computer architecture classes and younger students in person, explaining abstract concepts using familiar examples.
 - Communicate professionally with supervisors and families, responding quickly to feedback, scheduling changes, and clarifying questions.
 - Maintain a 100% 5-star review rate by adapting support to client needs, including clarifying problem specifications and responding flexibly to client requests.
-

Projects

Caiman Research Language - Undergraduate Researcher

Sept 2023 – May 2024

- Contributed to a research programming language and compiler targeting CPU/GPU coordination, working within a complex Rust-based build system and custom intermediate representation
- Developed and debugged benchmarks to validate language features (e.g., external function effects, arithmetic operations), improving test coverage and surfacing correctness issues
- Reverse-engineered project setup and build process in an undocumented codebase, identifying and resolving a missing dependency bug; authored onboarding documentation later adopted as project README
- Collaborated with researchers via Slack and code reviews to debug issues, understand system architecture, and iterate on language features in a shared codebase

FAT File System Implementation - Systems Project (C)

Apr 2024 – May 2024

- Re-implemented core components of a FAT-style filesystem in C, including block allocation, directory traversal, and file read/write operations, modeling on-disk data structures and access patterns
 - Implemented and validated file system invariants (e.g., allocation consistency, directory structure correctness) through targeted test cases and edge case analysis
-

Skills

Languages: Python, Java, C, OCaml

Tools: Git, Bash, Make, VS Code, Jekyll, HTML, CSS, GitHub Pages