Mitch Briles

+1 (949) 449-9262 | Salt Lake City, UT, USA | mitchbriles@gmail.com | mitchbriles.com

EDUCATION

University of Utah Salt Lake City, UT

B.S. in Computer Science and B.S. in Applied Mathematics *Aug 2023 - Dec 2026 (expected)*

- Coursework: scientific computing, machine learning, algorithms, embedded systems, data structures, numerical linear algebra, multi-variable calculus, real analysis, computer systems, probability & statistics, compilers, numerical analysis, interactive computer graphics, advanced OS
- Cumulative GPA: 3.88/4.00 | Dean's List (every semester)

WORK EXPERIENCE

Research Assistant Jan 2025 - Present

Regehr Compiler Group, University of Utah *Salt Lake City, UT*

- Working on compiler optimization and correctness under Professor John Regehr
- Co-author on "Translation Validation for LLVM's AArch64 Backend" at OOPSLA 2025
- Contributing to the LLVM project to fix miscompilations in compiler backends
- Building tools to verify compiler translations. Found various bugs in LLVM's RISC-V and AArch64 backends
- Giving a talk at the 2025 LLVM Developer Meeting: "Translation Validation for LLVM's RISC-V Backend"

CS and Math Tutor Jan 2024 - Sept 2025

Learning Center, University of Utah *Salt Lake City, UT*

- Guided hundreds of students to succeed in math and computer science

PUBLICATIONS

Translation Validation for LLVM's AArch64 Backend ⬈ OOPSLA 2025

Berger, R., Briles, M., Bushehri, N., Coughlin, N., Lam, K., Lopes, N. P., Mada, S., Tirpankar, T., & Regehr, J.

- Used LLVM to generate ARM assembly, lift back to IR, then verify the lifted function matches the original
- Invoked Alive2 to prove that compiled code is a refinement of the source
- Discovered 45 unique and previously unknown miscompilations in LLVM
- Modeled the semantics and execution environment of AArch64 code in LLVM IR
- Used advanced fuzzing techniques to generate tests

PROJECTS

Operating System Kernel in Rust

- Implemented a small operating system kernel in Rust targeting x86-64 systems
- System includes a memory allocator, ELF loader, address spaces, processes, context switching, and scheduling

SKILLS

- **Programming Languages**: C, C++, C#, Python, Java, R, MATLAB, Rust, x86-64, RISC-V, ARM, GLSL, +more
- **APIs / Libraries**: NumPy, PyTorch, TensorFlow, Qiskit, OpenGL, Vulkan, SDL, .NET MAUI, CUDA
- **Tools**: Git, Github, UNIX, Docker, Visual Studio, JetBrains IDEs, QEMU, Virtual Box, Nix
- **Other**: Drivers, binary exploitation, thread hijacking, DLL injection, jailbreaking, Raspberry Pi, robotics