Zane Enders

Salt Lake City, UT ▪ (801) 391-4150
zane.enders@gmail.com ▪ github.com/zaneenders ▪ linkedin.com/in/zane-enders

EDUCATION

University of Utah, Kalhert School of Computing Salt Lake City, UT
Bachelor of Science Candidate: Computer Science *December 2024*

- Undergraduate Research Position.
- Related Coursework: Compilers, Algorithms, Computer Systems, Linear Algebra, Programming Languages, Operating Systems, Models Of Computation, Distributed Systems (Fall 2024).

EXPERIENCE

University of Utah, Compilers and Programming Languages Lab Salt Lake City, UT
Research Assistant - Herbie Floating point Compiler *May 2023 – Present*

- Internal job server to allow more concurrent interactions to Herbie from its frontend Odyssey.
- Improved Regimes data layout (AOS to SOA) and algorithm which together lead to a 2–3× speed up.
- Upgraded the reports page from static pages to using JavaScript. This has allowed more interactive capabilities like sorting, filtration, and diffing against other reports, improving ease of development.
- Surfaced report metric (Bogosity) of data input quality.
- Weekly status update meetings with the team sharing progress and blockers.
- Attended the ARITH 2023 Conference in September 2023.
- Advised by Pavel Panchekha (University of Utah).

Contender Bicycles Salt Lake City, UT
Sales and E-Commerce *May 2014 – October 2021*

- Automated product page generation by building a web scraper.
  - Using Swift package SwiftSoup.
- Migrated 15,000+ products from Wordpress to Shopify.
  - First Swift project making a CSV parser to transform data formats.
- Over 1 Million dollars in personal sales for 2017 & 2018 (not commission based).

PROJECTS

- Compiler for an array programming language (Swift).
  - Implemented lexer, recursive descent parser, typechecker and x86 code generation.
  - Learned about LLVM, vectorization, e-graphs, LR parsing.
- Personal Website (Swift).
  - Built with Swift NIO and HummingBird.
  - DSL encapsulating HTML, CSS and minimal JavaScript (public release soon™).
  - Deployed on Fly.io.
- Scribe | End user experience model (Swift).
  - Declarative UI/UX DSL using resultBuilders, Observable and propertyWrapper.
  - Visual keyboard navigation over the abstract syntax tree.
  - Automatic starting selected Block.
  - Updates from input commands or async events.
- Chroma Shell | Unix terminal UI DSL (Swift).
  - Dynamic layout based on terminal size.
  - Built using Scribe.
- Linux HTTP server (Swift/ C).
  - Originally implemented using I/O multiplexing with epoll and UnsafeMutablePointers.
  - Basic resultBuilder DSL for generating HTML pages.

INTERESTS

- Skiing
- Mountain biking
- Compilers
- Operating Systems
- End-user programming
- Human Computer Interaction