Ian Li

Github: @ianayl, Linkedin: ianayl, Email: ianayl.work@gmail.com

Skills

Programming Languages	C, C++, CUDA, C#, Python, Java, Javascript, Typescript, Bourne Sh/Bash, Scheme, ARM assembly, R
Tooling	LLVM, Linux, Nix/NixOS, MS Azure, Selenium, Regex (PCRE), Make, Git, SSH, GDB/LLDB
Web Development	React, React Native, .NET Core, SQL, Express, SASS, Bootstrap, MongoDB, Flask
Certificates	Microsoft Azure AZ-900, Microsoft Azure AI-900

Experiences

Web Developer, *nvision* - *Markham, Canada* 2019

- Independently developed employee dashboards for visualization of metrics using **React** and **MongoDB** (**MERN** stack)
- Developed customizable Wordpress themes, collaborating with UI designers to bring design mockups into fruition
- Maintained **Linux** servers and MySQL databases to host Wordpress sites using the **LAMP** stack

Lead Software Developer, *Rentura (startup)* - *Toronto, Canada* 2021-2022

Lead the development and design of a *minimum viable product* for a B2C furniture rental platform

- Independently developed server backend in **Javascript**, with a **RESTful API** using **CRUD** endpoints secured with *tokens*
- Maintained **MongoDB** instances on a **Linux** server, and designed database schemas for an *order management system*
- Collaboratively developed responsive, mobile-first *e-commerce* frontends in **React.js** using **Sass** and **Axios** (**MERN** stack)

Ongoing Thesis

Improving KLARAPTOR: Automatic Finding of Optimal CUDA Kernel Launch Parameters Current

Automatically find near-optimal **CUDA** kernel parameters for fast GPU kernel execution. I am working on a(n):

- Method to detect constraints on kernel launch parameters using Scalar Evolution (SCEV) and Quantifier Elimination
- **LLVM** pass to find range of all valid launch parameters for a given kernel (to facilitate profiling using KLARAPTOR)

In order to speed up *data-oblivious* computations (*e.g.* matrix operations for machine learning/AI, computer graphics, *etc.*)

Projects

Experimental Language - repo Current

Shells are *outdated*: Creating a language in **C++** with goals to be a good general-purpose language *and* a fast replacement to shells

- Devised an automatic table-based **lexer generator**, and wrote a **recursive descent** parser following LL grammar
- Writing an interpreter as a **tree-walk evaluator**, and AST passes to generate CFGs **/ LLVM IR** for compilation
- Working on a static type system with type inference, goals for JIT compilation in future using LLVM

Mindless - repo 2021

Proof-of-concept **unconditionally secure** biometrics password manager based on hashing algorithms (SHA256)

- Developed a password manager using facial recognition landmarks (extracted using **OpenCV**) with a web frontend (**Flask**)
- Created a method to generate consistent passwords from facial landmark data: *passwords are* ***never*** *stored! Not even encrypted*
- Became a finalist project in a 36-hour hackathon (Hackwestern 8) as a one-man team out of 346 teams

Shell Site Generator (`shsg`) - repo Current *(On hiatus)*

Ultra lightweight and portable static site generator that will run on *anything* with a minimal UNIX-like environment

- Developing a static site generator using **POSIX shell** and **PCRE Regex**: *No dependencies other than a UNIX shell & coreutils!*
- Created a transpiler, transpiling markdown to themed webpages with no installation, compilation, or alternative runtimes required
- Working on removing dependency on PCRE extensions (support for Busybox), and parsing tables and lists in pure regex

Education

University of Western Ontario - *London, Canada* 2021 – Current

Candidate for Honours Specialization in Computer Science – Anticipated in Spring 2025, 3.9/4.0 GPA