Andrew Blinn

July 18, 2019

me@andrewblinn.com • github.com/disconcision
Toronto / Canada • English + French (basic)

Passionate about programming languages as user interfaces & keeps current with PL/FP/UI research & development

Work @ TodaQ Toronto

Software Development in Clojure & ClojureScript • *May 2019 - Current*
Around back - Helping implement a new distributed digital asset management protocol
Up front - Revamping & refactoring an SPA web portal in CLJS/reagent/re-frame + SASS

Work @ University of Toronto

Software & Documentation Development • *Summer 2018*
Wrote code & exegesis for professor David Liu's programming language theory course
Designed & built an educational language with pattern matching & algebraic data types

Teaching assistance for Principles of Programming Languages • *5 semesters, 2017 – 2019*
Led tutorials, managed fora & coached with a focus on TDD & code review
Built an algebraic code stepper to illustrate continuations in Scheme

Personal Projects

Fructure is a prototype editor focused on composable refactoring, featured at RacketCon 2019
Containment Patterns extend Racket pattern matching to capture contexts via composable continuations
Depthop is an OpenMP-parallelized C++ raymarcher featuring constructive solid geometry

Conferences & Workshops

Invited speaker at RacketCon 2019 • *Salt Lake City*
Spoke about structured editing in Fructure & connections to language-oriented programming

In attendance • *Salt Lake City, Toronto, Eugene, St.Louis*
2019: Racket Summer School, Clojure North. 2018: OPLSS, ICFP, Strange Loop, RacketCon

School @ University of Toronto

H.B.Sc in Mathematics & Computer Science • *2014 - 2019*
Graduate-level coursework in abstract algebra, compilers, graphics & languages
Built a Racket-based x86/C com/transpiler for a λ-calculus-based language with macro system
Coursework in algorithms, concurrency, differential geometry, operating systems & topology

Research in Variational Data Structures with Marsha Chechik & Ramy Shahin • *2018 - Current*
Built & profiled Haskell data structures supporting variational analysis of large software product lines
Designed & built SpyShare, a Graphviz-based tool to visually inspect data sharing

Research in Structured Editing in Racket with Gary Baumgartner • *Summer 2017*
Self-initiated study of existing refactoring, live programming & direct manipulation tooling
Designed & began implementation of Fructure, a Racket-based polyglot structure editor

Technical Skills

Functional Programming - Type/test-driven development in Racket, Clojure, Scheme & Haskell
Property-based testing with QuickCheck. DSL development with Racket/Redex

UI & Graphics - HTML, CSS, CLJS, React. Mock-up in Photoshop/Illustrator/+
Kinematics, raytracing & geometry processing in C++ & GLSL

Profiling & Parallelism/Concurrency - CUDA. C, C++ with OpenMP & MPI

More words - LaTeX, Python, Java, Git, Bash, GNU/Emacs/Linux/Windows/MacOS

Off-screen interests include bouldering, running, camping, taking pictures & bike-commuting year-round