

Gregory Schare

818 454 4044 · gs3072@columbia.edu · github.com/gschare · linkedin.com/in/gregory-schare

EDUCATION

Columbia University, New York, NY December 2023
B.A. in Computer Science and Mathematics. (GPA: 3.89)
Selected coursework: Language Design. Type Theory. Program Synthesis. Formal Verification. Analysis of Algorithms. Compilers. Cryptography. Algebraic Topology. Topology. Analysis. Algebra. Honors Math.

SKILLS

Programming Languages: Python, Haskell, C, OCaml, Rust, Bash, Java, Racket, JavaScript, TypeScript.

EXPERIENCE

Carnegie Mellon University Software and Societal Systems Department Pittsburgh, PA
Research Assistant, Composable Systems Lab May 2023 – August 2023

- Participant in NSF Research Experiences for Undergraduates in Software Engineering (REUSE) with Heather Miller
- Designed experiments to evaluate the performance of Collabs CRDT library in real-time collaborative rich text editing
- Developed benchmarking pipeline in TypeScript using Puppeteer, Chromium, and AWS EKS

Columbia University Department of Computer Science New York, NY
Research Assistant, Edwards Lab September 2022 – May 2023

- Designed optimizations for the reference counting algorithm of the `sslang` compiler
- Implemented IR tree transformations in Haskell using Scrap Your Boilerplate
- Attended weekly standups and collaborated with other working groups to resolve compiler bugs

CertiK New York, NY
Software Engineering Intern, Evolution Team June 2022 – September 2022

- Refactored legacy Python code of Autotest, a formal verification tool used to audit Solidity smart contracts
- Patched Autotest to prepare for verification of high-profile client contracts while following an agile methodology
- Designed improvements to Autotest's specification language, including streamlined syntax, loop and contract invariants
- Implemented translator from Solidity augmented with Hoare logic into z3 SMT solver expressions

Columbia University Center for Science and Society New York, NY
Research Assistant, Making and Knowing Project June 2020 – June 2022

- Improved textual analysis tool by refactoring legacy code and adding filters and new metrics for semantic similarity
- Achieved 77x speedup of textual analysis pipeline by upgrading parsing from regex to XML using Python lxml library
- Built lab websites using Google Drive API, Pandoc, and custom static site generators written in Haskell

PROJECTS

Dependently Typed Programming Language, *Types, Languages, and Compilers* with Raven Rothkopf

- Implemented the type checker for `slyce`, a dependently typed programming language written in Haskell

SMT-Powered Model Checker for LLVM, *Formal Verification* with Marcus Min

- Designed and developed `Thra11`, a bounded model checker for a subset of LLVM with CTL specifications
- Analyzed LLVM control flow graph using C++ and performed bounded model checking in Python using `cvc5`

Imperative Programming Language, *Programming Languages and Translators*

- Collaborated as a team of 5 students in order to divide responsibilities according to individual skills
- Designed and developed `josh`, a statically-typed imperative scripting programming language written in OCaml
- Implemented lexer, parser, semantic checker, and compiler using `ocamllex`, `ocamlyacc`, and LLVM

LEADERSHIP AND ACTIVITIES

Teaching Assistant, *Columbia University Department of Computer Science.*

Fall 2023 CSEE 6863 Formal Verification of Hardware and Software Systems

COMS 4995 Parallel Functional Programming

Fall 2022 COMS 4115 Programming Languages and Translators

Private Tutor. Tutor for COMS 1004 Introduction to Computer Science and Programming in Java during Fall 2023.

Interests: hiking, film, digital collage, writing, history of computer science, architecture.