Jan-Paul Vincent Ramos-Dávila

mail@jpramos.me | jpramos.me | jpramos-me | jpramos-me

EDUCATION

CORNELL UNIVERSITY

BS in Computer Science
BS in Philosophy
Expected May 2025 | Ithaca, NY
Conc. in Programming Languages
Conc. in Logic
College of Arts & Sciences

COURSEWORK

GRADUATE

Advanced Programming Languages
Runtime Verification
Category Theory
Software Testing

UNDERGRADUATE

Compilers + Practicum
Functional Programming
Modal Logic

SKILLS

LANGUAGES

Over 5000 lines:
OCaml • Scala • Python • LaTeX
Over 1000 lines:
Haskell • JavaScript • Java
Familiar:
C • Racket • Rust • Assembly

TOOLS

Unix • Git • VSCode • IntelliJ IDEA
Shell • Neovim • Docker • Heroku

HONORS

Winner, Third Place
ACM SIGPLAN Symposium POPL SRC 2023
Travel Scholarship
ACM SIGPLAN Conference PLDI 2022
Finalist, Mathematics
Regeneron International Science and Engineering Fair 2020 & 2021

ACADEMIC SERVICE

Student Volunteer
ACM SIGPLAN ICFP 2023

EXPERIENCE

CARNEGIE MELLON UNIVERSITY | Software Verification Research Intern

Advised by Dr. Jonathan Aldrich & Dr. Joshua Sunshine
May 2022 – Present | Pittsburgh, PA
Early researcher of Gradual Verification: a state of the art framework that leverages both static and dynamic verification to support incremental verification.

- Developed an algorithm to target Gradual C_0 predicates as second-order predicates, which significantly optimized runtime check assertions in our benchmarks from $O(2^n)$ to $O(n)$. This project was funded by Amazon.
- Exploring the application of gradual verification techniques to smart contracts on the *Algorand* blockchain platform in developing Gradually Verified Teal. This project was funded by Algorand.
- Worked on formal proofs for establishing semantic correspondence to ensure soundness between the static and dynamic verifier for Gradual C_0.
- Fixed optimization bugs and implemented a Property Based Testing tool for evaluating the soundness of Gradual C_0 source code.

CORNELL UNIVERSITY | Programming Languages Undergraduate Researcher

Advised by Dr. Adrian Sampson
October 2021 – December 2022 | Ithaca, NY

- Worked on a symbolic execution tool for verifying parallelism in Calyx.
- Implemented *Graphicionado Graph Analytics* algorithm in Calyx as a case-study of the language and found/solved soundness bugs in the front-end.

PUBLICATIONS

Evaluating Soundness of a Gradual Verifier with Property Based Testing Jan-Paul Ramos-Dávila In Principles of Programming Languages Student Research Competition (POPL 2023) (POPL Video) (POPL Poster) & In Cornell Undergraduate Research Journal (CURJ Vol. 2 No. 1).

PROJECTS

INCREMENTAL SPECIFICATION MINING Cornell CS 6156 Runtime Verification

- Instrumentation for Maven-based projects that *incrementally* mines specifications for runtime verification.
- Significantly decreases overhead for evolutionary-aware specification mining. Supports integration with Javert and BDDMiner.

ETAC Cornell CS 4120 Intro to Compilers

- Compiler for the Eta programming language in OCaml with ∼ 7,200 lines of code.
- Made use of Jane Street's expect testing suite for ∼90% code coverage.

RNAFOLDML Cornell CS 3110 Functional Programming

- OCaml package that enables users to input both RNA sequences in FASTA format and a set of constraints to predict RNA secondary structure.

DIFFEQ-LANG Senior High School Project

- Domain specific language for automatically solving and graphing differential equations via web interface.