Founding Software Engineer

Compilers, Programming Languages, & Verification

Highlights

- ⚲ Fully Remote
- ◉ ≥ $200k + ≥ 0.5%
- ⛓ Reports to CEO
- 📅 Unlimited Time Off
- 🏊 Paid Vacation Expenses
- 🖥 $4k Device Allowance
- ♥ Medical/Dental/Vision
- ⊪ Gym Allowance
- ☰ Learning Allowance

About Us

At **Formal**, we're rethinking serverless *from scratch*: we're building a new computing stack for instant, globally available, truly elastic, soundly isolated execution. We want anyone to be able to easily deploy their software on our machines and have it instantly react to inbound requests with bare-metal performance. We're leveraging formal methods to build OS interfaces and compilers for low overhead, formally verified isolation without containers or VMs. Our stack is simultaneously more secure and performant than anything else out there.

We believe in moving deliberately, with care, and breaking nothing. Incredible developer experiences govern how we think about the software we write.

Our Tech Stack

We write mostly in Rust, use `git`, prefer CLIs to GUIs and love OSS.

Equal Opportunities

Formal is an equal opportunity workplace. We understand, accept, and value differences amongst people and their backgrounds. We provide everyone fair opportunities to attain their full potential and actively promote a sense of belonging and wellness.

Expectations

You'll lead the development of a new, Rust-like programming language intended to run in kernel-space. Expect to write and review design documents, implementation decisions, lots of code, machine-checked proofs, and LaTeX specifications.

Responsibilities

- Develop an existing compiler codebase and software to continuously test and verify its correctness
- Formally verify language properties and implementations
- Collaborate and co-design with OS and networking teams to tightly integrate the language into the overall product
- Participate in both sides of code reviews
- Actively anticipate and communicate future roadblocks
- Participate in the open-sourcing of the language

Qualifications

- 3+ years of compiler software engineering experience
- Deep understanding of type systems and logic systems: formal specification, design, implementation, and trade-offs
- Familiarity with formal programming language semantic specifications, proofs, and ability to translate them into code
- Experience with formal verification including model checking, theorem provers and SMT-based approaches
- Understanding of system-level language concerns such as binary formats, loading, ABIs, linking, and machine targets
- Mastery of or ability to quickly master Rust
- High level of independence and autonomy
- Ability and willingness to document, teach, and explain complex technical problems and solutions

Compensation & Benefits

Compensation is comprised of a competitive market salary and stock grant conditioned on a vesting schedule. Benefits include unlimited paid and unpaid vacation time, reimbursable vacation expenses (transportation and lodging up to $1k), a triennial $4k computing device allowance, comprehensive medical, dental, and vision insurance, a $120 monthly gym allowance, and $250 yearly to spend on anything educational.