Tesla Zhang

ice1000kotlin@gmail.com · 🔗 https://ice1000.org · ⭘ https://github.com/ice1000

Education

B.S. in Computer Science at **The Pennsylvania State University**, PA, US   Aug, 2018 – Dec, 2022
Minor in Mathematics, GPA 3.28/4.00
Selected courses: Math 435 Abstract Algebra, Cmpsc 450 Concurrent Programming, Math 427 Foundations of Geometry, Math 429 Introduction to Topology, Math 437 Algebraic Geometry

Ph.D. in Computer Science at **Carnegie Mellon University**, PA, US   Aug, 2023 – Present
Advisor: Stephanie Balzer
Selected courses: 15-836 Substructural Logics, 15-791 Advanced Topics in Foundations of Types and Programming Languages

---

Work Experience

JetBrains Research, Remote   Jan, 2020 – Dec, 2020
HoTT and Dependent Types, Interactive Theorem Prover Development
- Used features like gradle composite build and buildSrc to reduce build time and improve automation.
- Improved the language/IDE, such as sections, hygiene macros, Fin type with elaborative subtyping, semantic highlighting, etc.
- Created an extensible REPL engine, provided implementations in CLI (with contextual completion using jline3) and in IntelliJ IDEA (interacts with the opened project, supports completion, highlighting and goto definition).
- Designed and implemented an expression type-checking debugger that supports step-into and displays local context and expressions as stack frames.

PLCT Lab, Remote   Dec, 2020 – Present
Implementation of Dependent Types, Opensource Maintainer
- Leading a team to explore modern techniques in type theory implementation, such as pattern unification, elimination of dependent pattern matching, Cartesian cubical type theory, termination check of recursive functions, phase distinction between layers of a type theory, etc.
- Organized reading activities on design and implementation of type theory, presented several talks in ∞-Type Café Summer School on Type Theory (in Chinese).

RisingWave Labs, Remote   Jul, 2022 – Jul, 2023
Streaming Database, Developer Intern
- Proposed an overhaul of the query plan AST design, which better facilitates the enum feature implemented in the Rust language.
- Implemented a pretty printing framework for trees with smart line fitting and Unicode art. Integrated into SQL explain.

Sourcebrella Inc., Shenzhen, China   Feb, 2018 – Jul, 2018
Static Analysis, Compiler Frontend, IDE Plugin Development
- Created IntelliJ/CLion/Eclipse plugin for the Pinpoint analyzer. Co-worked on the SonarQube plugin.
- Created a multi-threading cross Java/Kotlin source code indexer which can index Hadoop within 4 minutes.
- Learned a lot about Linux programming and the Clang/LLVM codebase.

PingCAP Inc., Remote   Aug, 2018 – Aug, 2019
Distributed Storage Systems, TiKV Intern – Ecosystem Team
- Improved many TiKV-relevant libraries, like optimizing the performance of grpcio, adding new features to procinfo.
- Helped to migrate the Protocol-Buffer library used by TiKV and its Raft implementation.
- Learned a lot about Rust programming, distributed system, working remotely, and databases.

Related Projects

Aya Prover, Practical Implementation of Dependent Types (role: project leader)   ⭘ aya-prover/aya-dev
- Supports dependent types, dependent pattern matching with confluence check for overlapping clauses, higher inductive types, GADTs (paper published), hierarchial universes, cubical type theory features, and implicit arguments.
- Can export elaboration result to HTML or LaTeX. Can JIT-compile closures into JVM using HOAS, Can refine patterns using coverage information. Supports both LSP in VSCode and IntelliJ PSI. Provide jlink binary releases.

IntelliJ Pest, Pest language plugin for IntelliJ Platform   ⭘ pest-parser/intellij-pest
- Semantic-based highlighting, completion, navigation, definition extraction/inlining, and Rust plugin integration.
- Provides live preview – test grammar files by dynamically highlighting user code according to the grammar on the fly. These highlighted code could be exported to HTML.

VSCode extension for Arend, Arend language server, based on lsp4j and Arend compiler's internals   ⭘ ice1000/vscode-arend

Arend IO, Experimental IO library for Arend, implements unsafePerformIO and simple IO actions   ⭘ ice1000/arend-io

---

Academic Experience

Multi-Campus REU, State College   May, 2019 – Aug, 2019
- Extensively studied the literature on row polymorphism and record calculus.
- Implemented (⭘ owo-lang/voile-rs) and formalized a row polymorphic dependent type theory.

Learning Assistant for Cmpsc 461 (Programming Language Concepts), State College   Aug, 2022 – Dec, 2022

Skills

- Programming Languages: multilingual (not limited to any specific language), especially experienced in Java Kotlin Rust C# Agda Haskell Arend, comfortable with Dart C C++ F# F★ Idris Perl MATLAB (in random order).
- Compiler: understand techniques like locally nameless, explicit substitution, ANF, (P)HOAS (in LF & logic programming), and NbE. Familiar with most parser generators, understand layout syntax parsing.
- Kotlin/Java: **10 years of experience**, familiar with JNI, JPMS, Gradle, Kotlin coroutines, and Swing.
- Type Theory: understand Martin-Löf type theory, coinduction, HoTT, and Cubical, familiar with Idris, Agda (**5 years** of experience, contributor), Arend and some Lean/F★/Coq.
- **JetBrains MPS**: understand concepts and applications of Language-Oriented Programming.
- IDE Tooling: **6 years of experience**, familiar with the IntelliJ Platform infrastructure (created Julia, DTLC, Pest, Kala Inspections, etc.), also have experience with VSCode plugin development.
- Mobile Development: **2 years of experience**, familiar with Flutter and Android.
- Tools: editor-agnostic, have experience with team tools like YouTrack, Jira, GitHub, BitBucket, Slack, JetBrains Space and more.

Misc

- Crates.io: https://crates.io/users/ice1000, publishing interesting Rust libraries
- IntelliJ Marketplace: https://plugins.jetbrains.com/author/10a216dd-c558-4aaf-aa8a-723f431452fb
- Languages: English - fluent (TOEFL 100), Chinese - native speaker
- Open-source contributions: https://ice1000.org/opensource-contributions, member of JuliaEditorSupport, agda, pest-parser, EmmyLua, arend-lang and more, contributed to **agda, Arend, libgdx, jacoco, KaTeX, shields.io, grpc-rs, intellij-solidity, intellij-haskell, intellij-rust, TeXiFy-IDEA, rust-analyzer** and other projects (apart from organization ones)
- StackOverflow: 6000+ reputation, also active on Proof Assistants (5000+ reputation) and other StackExchange sites
- Latest revision of this resume: one-page version https://tinyurl.com/y8xdlfug, complete version: https://tinyurl.com/y2v59t36
- Get the Chinese version of this resume: https://tinyurl.com/ya4urea8
- **1 dan** on CodeWars, ranked #111 on the whole site (Top 0.020%), solving and making new coding challenges primarily in Haskell, Agda, and Idris and some other JVM languages

Publications & Preprints

[1] T. Zhang, "A Simpler Encoding of Indexed Types," in *Proceedings of the 6th ACM SIGPLAN International Workshop on Type-Driven Development*, in TyDe '21. Republic of Korea: ACM, 2021. doi: 10.1145/3471875.3472991.

[2] T. Zhang, "Elegant elaboration with function invocation." [Online]. Available: https://arxiv.org/abs/2105.14840

[3] T. Zhang, "A tutorial on implementing De Morgan cubical type theory." [Online]. Available: https://arxiv.org/abs/2210.08232

[4] T. Zhang, "Three non-cubical applications of extension types." [Online]. Available: https://arxiv.org/abs/2311.05658

[5] T. Zhang, "Two tricks to trivialize higher-indexed families." [Online]. Available: https://arxiv.org/abs/2309.14187