

# TESLA ICE ZHANG

✉ ice1000kotlin@foxmail.com · ☎ (+86) 180-8192-5082 · 🌐 ice1000

## 🎓 EDUCATION

The Pennsylvania State University (PSU), PA, US

08/18 – Current

Major: Computer Science, Anticipated Graduation: 06/22

## 👨‍💻 WORK EXPERIENCE

Sourcebrella Inc., Shenzhen, China

02/18 – 07/18

*Intern* Compiler Front-End, IDE Plugin Development

- Created IntelliJ/CLion/Eclipse plugin of the Pinpoint analyzer, helped to create SonarQube plugin
- Created a multi-threading cross Java/Kotlin source code indexer, index Hadoop takes 4 minutes
- Learned a lot about Linux programming and clang/llvm

PingCAP, Remote

08/18 – Current

*Intern* TiKV Team

- Improved many libraries used by TiKV, like optimized the memory performance of grpcio, added new features to procinfo

## 🐱 PERSONAL PROJECTS

DevKt

<https://github.com/ice1000/dev-kt>

Cross-platform lightweight code editor / Kotlin IDE.

- Built-in Java/Kotlin highlights and completion, other languages can be supported via plugins (transplantable from JetBrains IDE's). Has extra build & run support for Kotlin.
- Flexible architecture, decoupled high-level logic with UI, migration to other UI frameworks is easy.
- Provides fine-grained highlight color and keymap settings, support hot reload.

Lice language

<https://github.com/lice-lang/lice>

Scalable programming language running on JVM.

- Supports lambda with and lazy (call by need)/macro (call by name)/strict (call by value).
- About 1/20 of Java's (Hotspot 8u151) performance, provides Java interaction and script engine support.
- Provides CLI REPL with GHCi-style code completion/colored output and JetBrains IDE plugin supporting semantic-based highlights, completion, go-to-definition, rename, eval-and-replace, quick fixes, etc.

Julia-IntelliJ

<https://github.com/ice1000/julia-intellij>

Julia plugin for JetBrains IDEs, compatible with all JetBrains Products.

- Semantic-based highlights, quick fixes, completion, parameter hint, special input mode for Unicode characters.
- Use Markdown plugin to highlight doc strings, provides REPL and SciView (showing the output of Plot libraries).

## ⚙️ SKILLS

- **Programming Languages:** **multilingual developer** (not limited to any specific language), and especially experienced in Java/Kotlin/Rust/C#/Agda/Haskell, comfortable with Dart/C++/F#/F\* (not listed in order)
- **Compilers:** familiar with various Parser Generators/Combinators, understand CPS and ANF, studied some theories about dependent types (like Luo's UTT, Martin-Löf Type Theory) and their implementations in programming languages
- **Kotlin/Java:** **2 years** of experience, **4** projects collected in Awesome Kotlin, familiar with JNI programming, the Gradle build tool, have experience in using the Kotlin compiler to analyze Java code
- **Formal Verification:** understand dependent types / (co)algebra and Curry-Howard Isomorphism, currently studying Homotopy Type Theory and Cubical Type Theory, familiar with Idris, Agda (**1 years** of experience, developer team member) and F\*

Source code studied: Agda, Idris, miniagda, Mini-TT

- **JetBrains MPS:** **1 years** of experience, understand concepts and usages of **Language-Oriented Programming**
- **IDE Toolings:** **2 years** of experience, familiar with the IntelliJ Platform infrastructure and overall architecture (created the Julia plugin and many others), also have experience with Eclipse/SonarQube
- **Mobile Development:** **2 years** of experience, Android (Java, Kotlin (Anko)), Fuchsia (Flutter)

- **Developing Tools:** can adapt to any editors/OSs, usually use JetBrains IDEs and Emacs under Ubuntu, have experience with team tools like YouTrack, Jira, GitHub, BitBucket, Coding.net, Tower

## ❖ MISCELLANEOUS

---

- Blog: <https://ice1000.org/> partially translated into English
- Literate Agda Blog: <https://ice1000.org/lagda/> used Literate mode of Agda improved by me
- Bintray: <https://bintray.com/ice1000> published some useful JVM libraries
- IntelliJ Plugin developer: <https://plugins.jetbrains.com/author/10a216dd-c558-4aaf-aa8a-723f431452fb>
- Books about formal verification I've written: <https://github.com/ice1000/Books>
- Languages: English - fluent (TOEFL 100), Chinese - native speaker
- Opensource contributions: <https://ice1000.org/opensource-contributions/> pull-requested (functional improvements) to organizations like *Microsoft*, *JetBrains*, *Ruby*, *Dropbox*, *PingCAP*, *TiKV*, projects like *agda*, *imgui*, *shields.io*, *intellij-solidity*, *IntelliJ-EmmyLua*, *intellij-haskell*
- StackOverflow: <https://stackoverflow.com/users/7083401/> 3000+ reputations, also active on other StackExchange sites
- Get the newest version of this resume: <https://tinyurl.com/y8xdlfug>
- **1 ky**u on CodeWars, ranked #37 on the whole site, primarily in Haskell
- Love making friends