

# KIVA Z.

✉ imkiva@islovely.icu · ☎ (+86) 178-8235-2997 · 🌐 imkiva · 🌐 <https://imkiva.com>

## EDUCATION

---

**Southwest Jiaotong University**, Chengdu, China 2019/09 - present  
Major in Physics, Undergraduate 2, GPA 3.83/4.00

## WORK EXPERIENCE

---

**PLCT**, Remote 2020/11 - present  
*intern* PL Researcher

**aya-prover** <https://github.com/aya-prover/aya-dev>

- A dependently-typed programming language.
- Implemented lexer, parser and part of typechecker.
- Helped with the coverage checking algorithm.
- Implemented pretty-printing and rust-like error reporting.
- Learning lots of type-theory related topics and implementations.

**Covariant Innovation**, Remote 2019/02 - 2020/07  
*Co-founder* Technical Leader

**CovScript Compiler** <https://github.com/imkiva/covscript-rs>

- CovScript compiler, which targets CovScript VM.
- Implemented gradual typing system with static type inference.
- Implemented an analyzer which checks possible runtime errors.

**Mozart++ Template Library** <https://github.com/libmozart/mozart>

- Implemented components, such as EventEmitter, String Formatter, Process, etc.
- Helped to separate modules and make framework highly decoupled.

## PERSONAL PROJECTS

---

**fluorine** <https://github.com/imkiva/fluorine-rs>

An advanced simply typed lambda calculus with Rust FFI support and various modern features.

- Trait-based object-oriented and operators overloading.
- Generic types with generic constraints
- ADTs and pattern matching.
- Optional type hints, bidirectional type inference and type checker.
- Easy-to-use ffi macros.

**KiVM** <https://github.com/imkiva/KiVM>

Simple Java Virtual Machine designed for embedded platforms, following JVM specification 8ed

- Support most of Java 8's new features, such as lambda expressions and Stream operations
- Garbage Collector using self-designed G1-Copying algorithm
- Standard JNI interfaces and bytecode control flow analysis
- Learned a lot about runtime optimization

## SKILLS

---

- **Programming Languages: multilingual** (not limited to any specific language), and especially experienced in Java/C/C++/Rust/Haskell/CovScript, comfortable with Kotlin/JavaScript/Swift (in random order).
- **Rust**: Familiar with Rust language features and implementations, especially its LLVM backend.
- **Kotlin/Java**: Know about HotSpot VM, GC algorithms and JIT compiler.
- **Proof assistants: Aya/Agda/Arend**: Learning dependent type and Homotopy type theory (HoTT). Participated in the core development of the Aya Proof Assistant.
- **Modern C++ (>= C++11)**: Have experience in templates and template metaprogramming, and understand the C++ implementation mechanism in LLVM.
- **Development tools**: Comfortable with any OS/editors, usually use JetBrains IDE, Vim under macOS and WSL, experienced in Git and GitHub.

## MISCELLANEOUS

---

- Language: English - CET-6, TOEFL-101, Chinese - native
- Open Source Contribution: <https://github.com/imkiva>
- **2 kyu** on CodeWars, primarily in Haskell
- Programming for interest, powered by love.
- Love math and music. Always willing to share. Prefer remote work