

Rajveer Singh Bharadwaj

rajveer.developer@icloud.com | github.com/Rajveer100 | linkedin.com/in/rajveer-singh-bharadwaj-5504b31a4 | [@rajveer_1000](#)

Algorithms . Compilers . Open Source

Experience

Apple Open Source - Swift, Swift Compiler Contributor | Remote | 5+ PRs

Dec '22 - Present

- Added unimplemented **expression cases** in the **ASTPrinter**. [GitHub> PR](#)
- Improved out-of-place **Binding Diagnostic** reflecting 'var' or 'let' **Binding Pattern**. [GitHub> PR](#)
- Improved **Cast and Coercing** Diagnostics when **Literals** are involved. [GitHub> PR](#)

LLVM Open Source, LLVM/MLIR Compilers Contributor | Remote | 20+ PRs

Sep '23 - Present

- [mlir] Rectify mishandling in **InsertOpConstantFolder** causing crash with assertion. [GitHub> PR](#)
- [clang] Disallow using "this" on **new** and **delete**. [GitHub> PR](#)
- [libc++] Fix ambiguity when using **std::scoped_allocator** constructor. [GitHub> PR](#)

Rust Open Source, Rust Compiler Contributor | Remote | 5+ PRs

Sep '23 - Present

- Switch from using **// ERROR** annotations with **-error-format** to error-pattern. [GitHub> PR](#)
- Bootstrap: Check validity of **-target** and **-host** triples before starting a **build**. [GitHub> PR](#)
- Remove all usages of **dont_merge** hack to avoid function merging. [GitHub> PR](#)

Fidelity Investments, Software Engineering Intern | FCAT | Bangalore

May '23 - Jul '23

- Analysed and Visualized **Mongo's** Full Time Diagnostic Data - **FTDC Metrics**. [GitHub> Gist](#)
- Tech Stack: Go . Mongo . AWS S3 . Grafana . Snowflake . Python . JavaScript . Git . Bash

Education

UG **B.Tech in Computer Science and Engineering**, Bangalore Institute of Technology | Bangalore, India

Jun '20 - Jun '24

HS **Higher Secondary Education**, Delhi Public School North Bangalore | Bangalore, India

May '18 - May '20

Achievements

Active **ICPC Regionalist Amritapuri Region**, '@MainActor', ACM International Collegiate Programming Contest (ICPC)

[Link>Cert](#)

Active **Solved over 5000+ problems in top programming platforms**, Competitive Programming

[Link>Profiles](#)

2022 **World Rank 1709 (AIR 389) in Google CodeJam Round 1C**, Google's Coding Competitions

[Link>Cert](#)

2022 **World Rank 1183 (AIR 857) in Google KickStart Round G**, Google's Coding Competitions

[Link>Cert](#)

2021 **World Rank 2833 in Meta HackerCup Round 2**, Meta's Coding Competitions

[Link>Cert](#)

Skills

PL/Compilers/iOS C++ . Swift . Rust . LLVM . MLIR . TableGen . FileCheck/Lit . Python . Java . Bash . Git . CMake . Ninja

Developer Tools Unix . macOS . Terminal . NeoVim . Xcode . JetBrains IDEs

Languages English (C2) . Spanish (A1)

Projects

MLIR Toy Compiler - MLIR/LLVM

Q1 - 2024

MLIR Mini Toy Compiler to support matrix operations, C++/MLIR/LLVM/TableGen

[GitHub>Link](#)

- Lowers a sequence of matrix operations with support for **tiling** and **bufferization**.
- Operations** include ConstantOp, AddOp, MulOp using **ODS (tablegen)**.
- Generates the **mini-opt** tool with **optimization** passes, partial/full lowering with **Affine**.

The Eva Programming Language - LLVM

Q1 - 2024

Built a programming language from scratch with LLVM Compilers Infrastructure, C++/LLVM

[GitHub>Link](#)

- Working on building a low-level programming **language** with **LLVM IR**.
- Production-level **language features** like functions/closures, lambdas and **OOPs** classes.
- Branch instructions, flow control and **code generation** with final **executable**.

Recursive Descent Parser- Tokenizers/Parsers/AST

Q4 - 2023

Implemented Recursive Descent Parser from scratch for a full programming language, Rust

[GitHub>Link](#)

- RDP Parser** built from scratch handling complex **constructs** during parsing process.
- Built **tokenizer** and parsing modules with generated **Abstract Syntax Tree (AST)**.
- Recursive** production rules to handle **generic functions**, implement **classes** and objects.

Crypto Tracker - iOS

Q4 - 2023

Visualise Live Crypto Price Charts With Comprehensive Overview, SwiftUI iOS App

[GitHub>Link](#)

- A Crypto Currency App tracking **live data** via third-party **API** with user portfolio.
- Dynamic **Chart Detail** utilising SwiftUI with **Combine** and **CoreData** for data **persistence**.
- Appealing visuals, **animations** and **core** functionality with great **UI** experience.

Competitive Programming - Templates

Q1 - 2023

Custom Competitive Programming Templates, C++/Rust Template Programming

[GitHub>Link](#)

- Implementation of **data structures** and **algorithmic** templates with C++ and Rust.
- Segment **Trees**, **Modular Integer**, **Combinatorics**, Disjoint **Set** Union and many more.
- Utilises **Generics** and **optimised** logic for fast **computations** in heavy input data.