RIDHAM KHURANA

khurana.ridham222@gmail.com | github.com/rmkhurana28 | 9466749330 | linkedin.com/in/ridham-khurana
Portfolio: www.rmkhurana.com

Education

Integrated M.Tech in Computer Science and Engineering (CSE) (Nov 2022 – May 2027)
University of Hyderabad, Telangana

EXPERIENCE

Summer Research Intern — Indian Statistical Institute (ISI), Kolkata (May 2025 – July 2025) **REPO**
Topic: Kyber and Dilithium Tweaks (Post-Quantum Cryptography)

- Implemented and analyzed cryptographic tweaks to the **Dilithium digital signature scheme**, including hash-function substitution, challenge coefficient expansion, and modified rejection sampling.
- Evaluated signature variability, performance, and verification success rates, ensuring compliance with **NIST post-quantum security standards**.
- Optimized and benchmarked the modified Dilithium implementation using **C (OpenSSL)** and **GCC 11.3 on Ubuntu**, achieving reproducible performance results.
- Contributed to a comparative study of **Kyber (KEM)** and **Dilithium (signatures)**, analyzing efficiency and robustness trade-offs.

Open-Source Contributions

GCC - GNU Compiler Collection

- Developed a GCC **-fanalyzer** patch for **getenv**, adding symbolic **NULL/non-NULL** bifurcation and improving diagnostics for unsafe environment-variable usage.
- Patch: https://gcc.gnu.org/pipermail/gcc-patches/2026-March/710912.html

QEMU — RISC-V Vector Engine (RVV)

- Fixed a **fractional LMUL legality bug in vsetvl** handling by correcting the **LMUL-vs-SEW divisor logic** in vector_helper.c, ensuring proper **RVV encoding semantics** for fractional LMUL configurations.
- PR: QEMU vsetvl fractional LMUL legality fix (Lore)

PROJECTS

RMc7 — Production C Compiler (Under Development) (Dec 2025 – Present) **REPO**

- Designing and implementing a **production-oriented C compiler architecture** in **C++17**, targeting **~90–95% ISO C language coverage**, with emphasis on modularity, extensibility, and long-term maintainability.
- Completed the **lexical analysis (tokenization)** phase, defining **~140 token types** and implementing a robust lexer with accurate **line and column tracking**.
- Establishing core **compiler abstractions** and **intermediate representations**, forming the foundation for parsing, semantic analysis, optimization passes, and x86-64 backend integration.
- Actively **re-engineering** the monolithic C implementation (RMc4) into a **clean, scalable C++ compiler pipeline**, suitable for incremental extension toward a full-spec compiler.

RMc4 — Complete C Compiler in C (Oct 2025 – Nov 2025) **REPO**

- Developed a **complete, end-to-end C compiler** from scratch in C, implementing core compiler phases including lexical analysis, parsing, semantic analysis, intermediate code generation, optimization, and **x86-64 assembly generation**.
- Supported a **well-defined subset of the C language**, including primitive data types **(int, char, double, bool)**, **arrays**, and control-flow constructs **(if / else, for, while)** with nested conditional logic.
- Designed and implemented a **recursive-descent parser** with operator-precedence handling, producing accurate **Abstract Syntax Trees (ASTs)** with **strong type checking**.
- Implemented **data-flow–based optimizations** using basic blocks and **control-flow graphs (CFG)**, including constant folding, **constant/copy propagation**, live-variable analysis, and **dead-code elimination**.
- Built a compiler with **robust error handling**, capable of detecting and reporting syntax, semantic, and type-checking errors with **meaningful diagnostics**.

RM-JobIn (MERN) (March 2025 – April 2025) **REPO**

- Developed a **MERN**-based job platform featuring **JWT authentication**, **role-based** application workflows, and **personalized tracking dashboards**.

SKILLS

Systems & Compiler Development

- Languages: C, C++ (C++ 17)
- Compiler Design: Lexical Analysis, Parsing, ASTs
- Middle-End: IR, CFGs, Data-Flow Analysis
- Backend: x86-64 Assembly, Optimization Techniques

Programming Languages

- Python, Java

Backend & Web Technologies

- Node.js, Express, REST, WebSockets
- React.js

Databases & Cloud

- MongoDB, SQL, AWS

Core Computer Science

- DSA
- Operating Systems
- Computer Networks