

# Paramuth Samuthrsindh

paramuths@gmail.com • paramuth@mit.edu • web.mit.edu/paramuth/www

*Résumé current as of March 4, 2026*

---

## Education

**Massachusetts Institute of Technology (MIT)** ..... **Class of 2026**  
*Candidate for B.S. in Computer Science & Engineering and Mathematics* ..... *Cambridge, MA*  
- Cumulative GPA: 5.0/5.0

---

## Experience

**MIT Parallel & Distributed Operating Systems Group (PDOS)** ..... **Sep 2025 - Present**  
*Undergraduate Researcher* ..... *Cambridge, MA*

- Built a benchmarking framework for agentic AI workflows covering GPU/CPU, memory, latency, energy, and cost, designed for extensibility and validated on RAG and OpenEvolve across multiple hardware accelerators.
- Introduced prompt reordering optimized for KV cache prefix caching, improving KV cache hit rate by 20% on the OpenEvolve workflow, directly reducing inference latency and time-to-accuracy.

**Amazon Web Services (AWS) – Glue Team** ..... **Jun 2025 - Aug 2025**  
*Software Engineer Intern* ..... *Bellevue, WA*

- Enhanced internal tooling for the Zero-ETL service by implementing DynamoDB query support and deep linking, automating a time-consuming manual workflow for engineers.
- Reduced first-request latency from over 10 seconds to less than 4 seconds by applying AWS Lambda SnapStart, compile-time dependency injection, and preload optimizations.
- Architected and implemented a Spark History Server management service that automated authentication, lifecycle control, and resource cleanup, enabling engineers to more efficiently visualize and debug Spark scripts.

**MIT Compiler Group (COMMIT)** ..... **Sep 2024 - May 2025**  
*Undergraduate Researcher* ..... *Cambridge, MA*

- Benchmarked parallel SpMV algorithms, including static, dynamic, and merge splits, on CSR matrices. The merge algorithm achieved up to a 6x speedup on 12 CPU cores over the serial baseline.
- Engineered a parallel interface in Finch, a Julia-to-Julia compiler, to support static and dynamic tensor computations across all storage formats, with performance comparable to TACO, a C++ sparse tensor algebra library.

---

## Projects

**Leiserchess NNUE** ..... **Feb 2025 - May 2025**  
- Developed an Efficiently Updatable Neural Network (NNUE) for Leiserchess, a chess-like game, to enable fast static evaluation of board positions.  
- Optimized evaluation by using an accumulator to incrementally update the first neural network layer and applying weight quantization to enable faster integer computation and better vectorization.

**Decaf Compiler** ..... **Feb 2024 - May 2024**  
- Collaborated on a team project to build a compiler for Decaf, a simplified version of C, handling the full compilation pipeline from tokenization to code generation.  
- Implemented key optimizations, including Copy Propagation and Dead Code Elimination using bit manipulation and graph traversal, and Register Allocation using a relaxed graph coloring algorithm.

---

## Selected Honors

**International Mathematical Olympiad (IMO)** ..... **2019, 2020**  
- Awarded a gold medal (2020) and a silver medal (2019)  
**King’s Scholarship (Thailand)** ..... **2021**  
- Selected as one out of nine King’s Scholars  
**William Lowell Putnam Mathematical Competition (Putnam)** ..... **2022**  
- Awarded an honorable mention

---

## Technical Skills

**Languages:** C, C++, Python, Rust, Go, Julia, Java, ASM, SQL, L<sup>A</sup>T<sub>E</sub>X, HTML, CSS, TypeScript, JavaScript, Smithy

**Tools:** PyTorch, LangGraph, LangChain, Hugging Face, vLLM, Git, Slurm, Unix, Docker, MongoDB, PostgreSQL, AWS