

# Ben Weinstein

Honours Mathematics Student, University of Waterloo  
bweinstein.me | LinkedIn | GitHub | brweinstein68@gmail.com | (613) 813-7617 |  
Citizenship: American / Canadian

## TECHNICAL SUMMARY

**Languages:** Rust, Python, TypeScript/JavaScript, C, HTML/CSS

**Frameworks:** Next.js, React, Node.js, Tailwind

**Libraries / Tools:** Tokio, Rayon, Clap, Pandas, NumPy, Matplotlib, Linux, Git

## EDUCATION

### University of Waterloo

Sep 2025 – Apr 2030

Honours Mathematics (Co-op)

**GPA: 4.0**

Relevant Courses: CS 135/136, MATH 135-138, STAT 230, PD 1, ECON 101/102

## EXPERIENCE

### Watonomous – Website Developer

Sep 2025 – Present

- Developed and maintained the team website using Next.js and modern frontend tooling
- Deployed production builds via Vercel and collaborated using Git-based workflows

### Event Executive – University of Waterloo Computer Science Club

Sep 2025 – Present

- Organized 10+ technical events and hackathons with 50–100 attendees each
- Coordinated with companies including Google, Atlassian, Stripe, and Zip
- Developed leadership, teamwork, and project management skills

## PROJECTS

### CLI Stock Visualizer

Python, Rust, Jupyter

- Built a modular CLI tool in Rust and Python for in-depth stock market analysis
- Implemented volatility metrics, return distributions, and risk analytics from historical market data
- Leveraged Rust (clap) for CLI parsing and Python (pandas, NumPy, matplotlib) for core computations
- Presented insights via a Jupyter notebook with LaTeX-formatted statistical explanations

### Terminal Email Client

Rust

- CLI and TUI email client written in Rust used by 50+ active users
- Implements an IMAP based inbox system
- Uses an SMTP client for sending mail through existing providers

### Executable Markdown Processor

Rust, Python

- Rust CLI tool executing fenced code blocks in Markdown and inserting outputs inline
- Used by 50+ Waterloo first year students with 15+ languages
- Features async execution, CLI parsing, Regex
- Refactor in Rust achieves 10,000× faster parsing than Python baseline.

### Interactive Cryptography Visualizer

Rust, TypeScript, React, WASM

- Interactive cryptography demos powered by Rust + WebAssembly and a Next.js UI
- Visualization and implementation of RSA, Diffie–Hellman, AES, SHA-256, and Discrete Log.

### Grep Clone from Scratch

Rust

- Reimplemented GNU grep with parallel file traversal, achieving equivalent speed and memory usage.
- Added regex, case-insensitive search, and syntax highlighting with a Clap-driven CLI.

## OTHER EXPERIENCE

### Habitat for Humanity – Restore Volunteer

- Managed inventory, tested electronics, and supported team logistics, developing teamwork and organization skills
- Contributed 250+ volunteer hours

### Canadian Forces Morale and Welfare Services RecPlex / East Lake Shores – Lifeguard

- Ensured safety and supervised recreational activities, demonstrating responsibility and reliability (2022–2025)