

# Constantin-Razvan Maracine

GitHub | mrazvan.dev | razvan.maracine@protonmail.com

## ABOUT

---

I am very enthusiastic about programming and science, with a focus on systems programming and security. Most of my free time is spent programming what fascinates me (e.g. TCP / IP protocol, emulators, etc.). I do not have previous work experience but where I lack in experience I make up for it with enthusiasm and perseverance.

## EDUCATION

---

**Colegiul National Alexandru Lahovari, Romania** *September 2020 - June 2022*  
*Natural Sciences*

**Colegiul National Mircea cel Batran, Romania** *September 2022 - June 2024*  
*Computer Science & Mathematics*

**University of Bucharest, Romania** *September 2024 - Expected July 2026*  
*Computer Science & Mathematics, Olympiad class*

## SKILLS

---

**Programming Languages:** C, Rust, Go, Zig, Python, Bash, C++, JavaScript, TypeScript, HTML, CSS, SQL

**Technologies:** MongoDB, Linux, Nix, NixOS, Docker, Git, GitHub, Just, Make

**Languages:** CAE English C1 certificate

## AWARDS

---

Qualification to the National Olympiad in Informatics, (*top 70*) *2023*

3rd place in MateInfoUB, *University of Bucharest* *2023*

Won the N-able Hackathon organized by ITFest, SISC, *Bucharest, Romania* *2023*

## PROJECTS

---

You can find more projects on my GitHub, here are some of the most interesting ones:

**Chip8 Emulator** *July 2024*  
A Rust implementation of the chip8 console instruction set using a graphical library for rendering. Contains unit tests for various opcodes.

**MDB** *September 2024*  
A minimal markdown parser written in C that uses no other external dependencies other than libc. MDB transforms your markdowns into blog posts to be integrated in your static website.

**CPN** *July 2023*  
CPN is a CLI written in Go for automating the process of downloading / running tests from various competitive programming online judges, being integrated with 'competitive companion' browser extension. Has other quality of life commands.

**Conway Asm** *December 2023*  
Implementation of Conway's Game of Life using assembly x86\_64 with some twists. It is part of the 'Computer Systems Architecture' university course and the solution has been stress tested against C / C++ naive solutions, with tests generated using Rust.