Tarek Hireche

+1 438-389-2094 | Montréal, QC | tarik.hireche@engineer.com | github.com/htarek-bytes | linkedin.com/in/hirechetarek | htarek.dev

Education

Université de Montréal Montréal, QC
Bachelor of Science, Computer Science *Jan 2022 — Apr 2026*

- **Cumulative GPA**: 3.41/4.3 | DIRO Excellence Scholarship (Apr 2024)
- **Relevant Coursework**: Operating Systems (C), Computer Architecture (MIPS), Microprocessors, Compiler Design, Memory Management, Data Structures & Algorithms.
- **Achievements**: Top 30 NorthSec Cybersecurity CTF (May 2025).

Technical Skills

- **Systems & Low-Level**: Zig (Proficient), C++ (STL/Algorithms), C (Academic), MIPS Assembly, VHDL.
- **Embedded Concepts**: Manual Memory Management (Stack/Heap), Pointers, Interrupts, GPIO, Recursion.
- **Debugging & Tools**: GDB, x64dbg, Valgrind, Make/CMake, Linux/Bash, Git, LaTeX/Typst.
- **Languages**: English (Native), French (Native), Arabic (Native), Spanish (Basic).

Projects

Zig, Systems Programming, Custom Memory Allocator (github.com/htarek-bytes/UdeM-IFT2035-HM2) Jan 2025 — Present

- Engineered a manual memory allocator replacing the standard library to achieve O(1) allocation speed.
- Implemented custom "Pile" (Stack) structures and memory recycling to manage heap memory without garbage collection.
- Demonstrated deep control of memory alignment, pointer arithmetic, and cache locality.

C++, Dynamic Programming, Algorithmic Optimization Engine (github.com/htarek-bytes/UdeM-IFT2125-Assignment1) Fall 2024 — Mar

- Implemented high-efficiency solutions for NP-hard problems (Knapsack, Coin Change) using Greedy Algorithms.
- Designed advanced data structures (Self-balancing Trees) to optimize time complexity for large datasets.
- Applied rigorous Big-O analysis to reduce runtime, mimicking constraints of embedded environments.

Assembly, Bare Metal, MIPS32 Matrix Validator (github.com/htarek-bytes/UdeM-IFT1227-D3) Oct 2024 — Dec 2024

- Developed a raw assembly program to validate "Magic Square" matrices, operating directly on CPU registers (`$t0-$t9`).
- Manually managed the call stack and frame pointers (`$sp`, `$fp`) to implement recursion without high-level abstractions.

Prolog, Compiler Theory, System F Interpreter (github.com/htarek-bytes/UdeM-IFT2035-HM3) Nov 2024 — Dec 2024

- Built a polymorphic type checker for the Girard (System F) functional language.
- Implemented capture-avoiding substitution and alpha-equivalence checks to verify type correctness.

Work Experience

Academic Tutor (Computer Architecture & Math) Jan 2024 — May 2024
UdeM / Math Plus *Université de Montréal*

- Mentored students in Computer Architecture, simplifying hardware concepts like CPU cycles and MIPS pipelines.
- Taught discrete mathematics and calculus, helping students decompose abstract problems into logical steps.

Technical Ambassador Jan 2024 — Present
Cap Campus (UdeM) *Montréal, QC*

- Presented Computer Science principles to high schools, demystifying algorithms and promoting STEM careers.
- Translated high-level technical domain knowledge into accessible presentations for non-technical audiences.

Technical Support Representative May 2022 — Feb 2024
Gatestone & Co. *Montréal, QC*

- Diagnosed complex connectivity issues and enforced strict authentication security protocols for enterprise clients.