

JEAN-FRANÇOIS TO

☎ 438.876.0741 • ✉ jeanfrancoisto@outlook.com • 🌐 github.com/jfto23 • jfto23.github.io

EDUCATION

McGill University • Montreal

Expected Graduation in Spring 2022

Bachelor of Science • Mathematics and Computer Science • GPA: 3.94/4.0

AWARDS

- Ministry of Higher Education's Excellence Bursary for Computer Science

TECHNICAL SKILLS

- Programming languages: Python, Java, JavaScript, Rust, C, Bash
- Other related experience: Git, Linux, SQL, VueJS, NodeJS, Flask, Travis CI, Jenkins, Docker, HTML, CSS

WORK EXPERIENCE

Software Developer Intern – Nakisa

May 2021 – August 2021

- Used the Spring Framework to build infrastructure/automation for a financial application
- Designed and created high-performance parsing and data processing algorithms for large amounts of JSON data
- Wrote and optimized scripts to automate data entry/validation that was previously done by a team of three
- Used Jenkins for automatic building and deployment of the application

Tutor – McGill University

January 2021 – Now

- Tutor for McGill's Computer Science Undergraduate Society (CSUS)
- Held weekly office hours to help undergraduate students one-on-one
- Tomlinson Engagement Award for Mentoring recipient
- Courses include: COMP202, COMP206, COMP250, COMP251, COMP273, COMP302

Tutor – Collège Jean-de-Brébeuf

August 2017 – May 2019

- Tutored students in various freshman-level courses such as math, physics and chemistry
- Helped students one-on-one to further their understanding of the subject

Research Assistant – Collège Jean-de-Brébeuf

Summer 2019

- Worked in a laboratory researching inorganic chemistry
- Wrote new labs so they could be taught to future students

PROJECTS

For more details about the projects, visit my [personal website](#)

Full Stack Website for Touch Typing

- Built a website that lets users play a typing game and compare their score with other players
- Features over 1000 texts and keeps track of high scores for each of them
- Languages/Tools: MySQL, NodeJS, ExpressJS, VueJS, Docker, Heroku

Multiplayer Game

- Built a video game with Rust using an open-source game framework (ggez)
- Designed and built a UDP peer-to-peer network from scratch
- Used WebAssembly to make the game playable on a browser

Chip-8 Interpreter

- Built a Chip-8 interpreter with Rust
- Capable of playing basic games such as Tetris, Space Invaders and Pong

Module for Open-Source Status Bar

- Wrote a Rust script that parses .ics files and shows the current and next event of the day
- Can be integrated as a module for an open-source status bar (Polybar)