Nabil Mansour

n1mansour@torontomu.ca | nabilmansour.com/ | linkedin.com/in/nnym/ | github.com/NabilNYMansour

Education

Toronto Metropolitan *(formerly Ryerson)* University Toronto, ON
B. Sc in Computer Science with a minor in Philosophy *2019 – 2024*

- Advisors: **Dr. Marcus Santos**, **Dr. Tim McInerney**
- **3.92/4.33** CGPA

Experience

Research/Teaching Assistant May 2021 – May 2022 & Sep 2023 – Jan 2024
Toronto Metropolitan (formerly Ryerson) University *Toronto, ON*

- Developed an **auto-marking** program in **LISP** that runs student programs and grades them automatically while also reporting any problems and handling any raised errors in their programs.
- Redesigned and modified CPS 305 (**Data Structures**) labs by improving on the instructions of the first drafts of the labs and providing solutions for them as supervised by **Professor Marcus Santos**.
- Administered weekly tutorials, critiqued students' code, and provided guidance for writing more efficient and readable code by discussing with them good standards and practices.
- Also TA-ed in other courses like CPS 506: **Comparative Programming Languages**.

Software Engineer May 2022 – Aug 2022 & Jan 2023 – April 2023
Fairly AI *Toronto, ON*

- Developed a reusable **pip-installable** Python library for enhanced code maintenance and future project extensions.
- Implemented big data verification functions that evaluate the potential risk of datasets and AI models.
- Optimized **Streamlit** applications and **Jupyter** notebooks for dynamic compatibility with diverse datasets and AI models, improving user experience.
- Redesigned the database for the back-end, and implemented a secure authentication system utilizing **Oso**, resulting in enhanced data integrity and improved system security.
- Created a user department system using **React** and **Flask** for efficient organization within companies.
- Restructured Data Validation processes, utilizing **Dask** to manage Big Data efficiently, resulting in improved accuracy and processing speed.

Projects

FractiX and Fractal Glide | *C#, HLSL, Unity Game Engine, MonoBehaviour, Steamworks* **Website** | **GitHub**

- Developed an **open source** cone/ray marching engine called **FractiX** from the ground up as an extension to the **Unity3D** rendering pipeline, optimizing rendering performance and achieving stunning fractal visual effects.
- Conceptualized, designed, and developed **FRACTAL GLIDE** an indie game available on **Steam** as a sample use for this rendering engine.
- Documented the process of development in my **YouTube channel**.

Neuro Gambit | *Python, Pytorch, Jupyter Notebook* **GitHub**

- Developed Chess Artificial Neural Networks, using **PyTorch**, trained on datasets from **Kaggle** and **FICS** games.
- Utilized methods used to avoid overfitting like **L2 regularization**, **dropout neurons** and **early stopping.**

Slime Simulator | *Python, ModernGL, GLSL, imgui* **GitHub**

- Implemented a slime mold cellular automata simulation that showcases the emergent behaviour of slimes.
- The purpose of this project is to learn how to utilize compute shaders in an effective manner and be able to use them in a rendering pipeline.

Self Parallel-Parking Arduino Car | *C++, Arduino* **GitHub**

- Developed the software for an **embedded system** that allowed a robotic **Arduino** car to parallel park on its own given a 35cm by 20cm parking slot that is between two parked objects.
- Received a grade of **10/10** as part of the final exercise for CPS 607: **Autonomous Mobile Robotics** course.

Technical Skills

Languages: Python, TS/JS, HTML5, CSS, C/C++/C#, GLSL, HLSL, LISP, Java
Frameworks and Libraries: Pytorch, Numpy, Pandas, Dask, OpenGL, DirectX, ModernGL, Node.js, React, Material UI, Cypress, Redux, Socket.io, Flask, SQLalchemy, Oso
Developer Tools: VS Code, Visual Studio, Git, Arduino, Emacs, Linux
Applications: Unity Game Engine, ShaderToy, MATLab, Fusion360