

Patrik Buhring

🐙 [OptimisticPeach](#) | ✉ pbuhring@uwaterloo.ca | ☎ [+1 \(647\)-460-7000](tel:+16474607000) | 📱 [Patrik Buhring](#)

Education

University of Waterloo, Canada

Sept. 2022 – Aug. 2027

- Double Majoring in **Computer Science** (91% GPA) and **Pure Mathematics** (86% GPA).
- Candidate for a **Bachelor's Degree in Mathematics**.

Experience

Informatics Coop Student

Jan. 2024 – Aug. 2024

University of Guelph – Center for Biodiversity Genomics

Go, C, Python, and Bash

- Drove a genetic sequence clustering project to completion, researching, and validating the pipeline.
- Optimized pipeline to achieve **40-80% memory reduction** and across-the-board runtime reduction.
- Implemented, and validated multiple genetic and geospatial aggregation endpoints using **FastAPI**.

Projects

Terminal Game Engine

Dec. 2023

C++

- Wrote a **terminal game engine** as well as a space invaders style game and a snake clone.
- Designed an incredibly flexible **ECS** API that achieves **realtime framerate** on a **terminal**.
- Incorporated **mouse input**, fine-grained rendering using braille characters, and colour support.

[Hypersphere: An Exploration of 3D and 4D Spherical Geometry](#)

Dec. 2022 – Present

Rust, WGSF, WebGPU, WASM, JavaScript, HTML, and CSS

- Publishes updates to [a live WebGPU enabled demonstration](#) in a custom-written HTML website.
- Integrates a custom flat shading water shader and various terrain generation modes for visual appeal.
- Developed an open source [4D math utilities](#) library and [separate tech demo](#).
- Adapts an existing [hydraulic erosion simulation](#) for spherical terrain: [demo video](#).

[Hydraulic Erosion Simulation on Spherical Terrain](#)

May 2023 – Present

Rust

- Adapts an existing [hydraulic erosion simulation](#) for spherical terrain: [demo video](#).
- Presented a talk demonstrating the project at the Summer 2023 SASMS at UWaterloo.
- Leverages a **SIMD** implementation of simplex noise and multithreading to efficiently use resources.

[Hexasphere: Open Source Sphere Generation](#)

Aug. 2020 – Present

Rust

- Implements an efficient sphere subdivision algorithm with the aim of reducing distortion.
- Over [2.7 Million downloads](#), with more than [16000](#) projects depending on Hexasphere on GitHub.
- Maintains and updates the project, ensuring quality and well-documented code.
- Optimized to produce cache-friendly meshes for efficiency when rendering very detailed spheres.

Skills

Languages Rust, Go, C++, C#, Java, C, Bash, Dart, Python, Racket, HLSL, GLSL

Frameworks & Libraries OpenGL, DirectX 11, Android, .NET, Linux, CUDA

Spoken & Written Languages English, Spanish, French