Arnav Kumar

a8kumar@uwaterloo.ca arnavcs arnavcs.github.io Arnav Kumar

Skills

Programming: C++, C, Python, Rust, TypeScript, GLSL, Go, Haskell, Bash
Tools: WebGL, Three.js, Godot, Unity, Unreal Engine, Matplotlib, Pytorch, Scikit-learn, OpenCV, Unix

Education

Candidate for BSc. Computer Science (4th year) @ University of Waterloo *Waterloo, ON.* | 2022 - Present
- 4.0 GPA, all advanced level courses, and term distinction. Received four merit based scholarships totaling $46,000.

Experience

Graphics & 3D Geometry Software Engineer @ Arcol.io *San Francisco, USA* | Sep 2025 - Dec 2025
- Designed and implemented a Rust algorithm to reconstruct terrain meshes from raw contour data for areas over 5km^2.
- Performed prototyping, iteration, and proof-of-concepts to choose and implement the most suitable approach.
- Rendered generated terrain and contour lines, sent over the Wasm ABI to TypeScript, with Three.js and WebGL.

Software Engineer @ Trend Micro *Kanata, ON.* | Jan 2025 - Apr 2025
- Engineered utility functions for Alicloud's webservices, optimizing cloud workflows and securing data management.
- Developed serverless functions to send scan metrics and results to the backend, helping identify details about customer errors and significantly cutting customer costs.

Combinatorics Researcher @ University of Waterloo *Waterloo, ON.* | May 2024 - Aug 2024
- Authored a paper (The dimension of sparse random graph orders) in pre-print in arXiv (https://arxiv.org/abs/2504.19029).
- Worked on novel research exploring random graph orders and poset dimension utilizing configuration models, approximation techniques, convergence in probability, devising constructions, and computationally searching for counterexamples.

Blockchain Software Engineer @ Dandelion Networks *Remote* | May 2023 - Aug 2023
- Developed and implemented an enhanced lattice syncing and node discovery algorithm in Go, querying peers with Protocol Buffers to identify and address missing blocks in the local lattice with secure and concurrent updates.
- Created Jenkins pipeline which automatically build, test, vet, and format pushed code to catch regressions.

Projects

4D Raymarching Pathtracer | *C++*
- Implemented SDF raymarching in 4D space with CPU pathtracing to contruct GIF renders of the scene.
- Encoded the render such that every frame in the GIF, the camera sends out rays into a 3D affine subspace of the 4D space, which are scattered into 4D space upon colliding at a surface.
- Designed and wrote a powerful material system with microfacet BRDFs, glass, volumetric fog, and the ability to assign a geometry more than one material.

Rigid-body Particle Simulation | *C++*
- Simulated gravity, spherical constraints, and collision physics for rigid body particles.

Software Rasterizer | *C++*
- Recreated the whole rasterization process on the CPU, including perspective correct interpolation for textures.

Software Raytracer | *C++*
- Implemented the Möller–Trumbore algorithm for fast ray-triangle intersection
- Programmed Lambertian diffuse, specular refraction, and specular reflection behaviour, and support for spherical environment mapping

Renovating the Labyrinth (Game) | *JavaScript*
- Solo submission for the 72 hours Waterloo game jam; voted winner of the technical achievement award.
- Built a real time CPU ray caster with ordered Bayer matrix dithering, rigid body collision, and random maze generation.

Achievements

- 335th (top 10%), with score 26, **William Lowell Putnam Mathematical Competition** 2022
- 10th in Canada, **Asian Pacific Mathematics Olympiad (APMO)** 2022
- 18th in Canada, 2 time qualifier, **Canadian Mathematical Olympiad (CMO)** ..... 2021 - 2022