Daylyn Nguyen

604-401-1805 | dkn9@sfu.ca | Coquitlam, BC | LinkedIn | Portfolio | github.com/daylyn-n

Education

Simon Fraser University — Burnaby, BC
Bachelor of Science in Computer Science — *2025 – 2029*
Relevant Course Work: Data Structures and Algorithms, Software Engineering, Linear Algebra

Douglas College — Coquitlam, BC
Associate of Science, Dean's List — *2024 – 2025*

Projects

Real-Time Lightning Visualizer | *C++, OpenGL, WebSockets, Concurrency* — Jan 2026 – March 2026 | GitHub
- Built a real-time lightning visualizer using C++, OpenGL as the renderer and ixwebsockets to get lightning strike data
- Used OOP diesgn to implement methods to render and load textures and lightning strikes data in a scalable way
- Utilized mutexes to parse through lightning strike data while rendering the points without running into race conditions

Particle Simulator | *C, SDL2, Physics, Spatial Partitioning* — Feb 2025 – Feb 2025 | GitHub
- Developed a real-time 2D particle physics simulator in C using SDL2 for rendering, and input
- Implemented elastic collision detection and impulse-based resolution using vector math, collision normals, and conservation of momentum
- Optimized collision detection using a uniform grid spatial partitioning system, reducing pairwise checks and improving performance by 20fps
- Used CPU profiling tools like valgrind and callgrind to debug non-obvious bottlenecks, reducing CPU usage by 40% and gaining 50fps

Maze Generator | *C++, SFML* — Dec 2025 - Jan 2026 | GitHub
- Engineered an interactive maze generator using recursive backtracking (DFS) with real-time rendering
- Designed modular architecture separating grid state, algorithm logic, and rendering systems
- Implemented stack-based backtracking and bitwise wall encoding to optimize memory efficiency

Rubik's Cube Solver | *Java* — Nov 2025 – Dec 2025
- Modelled cube state using cubie permutations and orientations parsed from input files
- Implemented Pattern Databases, BFS, IDA*, and heuristic search to compute near-optimal solutions
- Optimized memory usage for pattern databases and reduced solve length from 30+ moves to under 14

Clubs & Involvement

Competitive Programming Club — Burnaby, BC
Member, Simon Fraser University — *2025 – Present*
- Practicing algorithmic problem solving once a week involving graphs, dynamic programming, and data structures
- Strengthening proficiency in C++ coding challenges and contest-style problems

Game Development Club — Burnaby, BC
Member, Simon Fraser University — *2025 – Present*
- Developing self-directed graphics and game programming projects using C/C++, OpenGL, SDL2, and raylib,
- Implementing rendering pipelines, input systems, and real-time simulation features to deepen low-level graphics knowledge
- Engaging with peers to exchange technical feedback, explore engine architecture concepts, and stay accountable to long-term projects

Technical Skills

Languages: C++, C, Java, Python, JavaScript, LaTeX, HTML, CSS
Systems & Tools: Git, GitHub, Linux (WSL), GCC, Clang, Valgrind, Callgrind, gdb, CMake, Neovim, Tmux, VS Code
Libraries & Frameworks: OpenGL, Vulkan, SDL2, ImGui, Raylib, SFML, NumPy, Pandas, Matplotlib, SQL
Core Concepts: Data Structures, Algorithms, Graphics, TCP Networking, Concurrency