Nathan Hutton

LinkedIn : www.linkedin.com/in/nathanhutton/ Email : nathan.d.hutton@gmail.com
Portfolio : nathan-hutton.github.io/Portfolio GitHub : github.com/Nathan-Hutton

Education

University of Utah SLC, UT
Master of Computer Graphics and Data Visualization — GPA: 4.0 *Dec. 2025*

Westminster University SLC, UT
Bachelor of Computer Science — GPA: 4.0 *May. 2024*

- Applied mathematics minor.
- Received the outstanding computer science student of the year award.

Salt Lake Community College SLC, UT
Associate of Science — GPA: 4.0 *May. 2021*

- Earned degree while in high school.

Experience

Flight Safety International Broken Arrow, OK
Software Engineer Intern *May. 2025 - Aug. 2025*

Idaho National Laboratory SLC, UT
Software Engineer Intern *Feb. 2023 - May. 2024*

- Debug and reformat C++, C, and Python code.
- Work with waveform code on Linux.
- Convert Matlab test harness to Python.
- Configure CI/CD pipelines.
- Implement GUIs which interface with a backend.
- Converted documents made in LibreOffice to Latex documents.

Westminster University SLC, UT
Computer Science Tutor/TA *Aug. 2022 - Dec. 2023*

- Guide computer science students through computer science and algorithms classes.

Westminster University SLC, UT
IT Technician *Aug. 2021 - May. 2024*

- Help Westminster students and faculty with technical issues over the phone and in person.

Projects

- **Boids Flocking Simulation** [**Demo Video**]: OpenGL flocking simulation in C++. A simple ruleset by Craig Reynolds leads to complex flocking behavior. Uses IMGui, color blending using HSB colorspace, and obstacle avoidance.
- **Solar System** [**Demo Video**]: Solar system simulation with dynamic physics in OpenGL/C++. Utilizes shadow mapping, HDR, environment mapping, verlet numerical integration, bloom, and more.
- **AQI Visualization** [**Demo Video**]: Interactive visualization for Utah's air quality index using Javascript and D3. I made 68% of all commits in this group project.
- **Squibblets**: 4-person team Unity project using AGILE. I made 54% of all commits and created the main gameplay loop, movement, online leaderboards, audio, AI, UI, menus, and more.
- **Ray Tracer**: Python Ray Tracer made with no reliance on third party libraries.
- **Minijava Compiler**: Includes type-checking, lexing and parsing with ANTLR4, JVM assembly code generation, AST generation, flow control, package inclusion, and method calls.
- **Java Chatroom**: Terminal based chatroom utilizing java.net. Features include broadcast messages, private messages, user list retrieval, and notifications when users join or leave.

Programming Skills

Languages: C++, Python, C, Java, C#, Javascript **Tech**: OpenGL, Linux, Docker, Gitlab CI/CD, D3