

**Tomas Stegemann** / [sunsite.pages.dev](https://sunsite.pages.dev) / [stege084@umn.edu](mailto:stege084@umn.edu)

Bachelor of Computer Science, expected May 2024, GPA 3.7, Minor in Interdisciplinary Design  
College of Science and Engineering, University of Minnesota-Twin Cities

## Skills

C, C++, C#, Python, OpenGL, GLSL, Unity, Blender, RenderDoc, Git, Make, GDB, Valgrind, concurrent and parallel programming, graphics programming, game development, interdisciplinary design

## Work Experience

### Game Developer, College of Design (August 2023 - December 2023)

*Environment: Unity, C#, OpenGL, C++, GLSL*

Served primarily as a tools programmer, shader programmer, and programmer/designer for all systems and gameplay. Developed an educational game for teaching course material about trend forecasting to design students, as well as bespoke tools to create content for the game. Communicated with professors and students in the school of design to shape and refine the project according to their needs.

### Undergraduate Research Assistant, Interactive Visualization Lab (November 2022 - June 2023)

*Environment: Unity, C#*

Served primarily as a shader programmer and simulation/interaction programmer. Developed an interactive voxel/mesh hybrid canoe carving simulation to give indigenous communities a platform for sharing their expertise. Gave interactive demos of the simulation and employed user feedback in improving it. Collaborated closely with other lab members so that my work could be integrated into the lab's overarching projects.

### Software Development Intern, Advaita Bioinformatics (June 2019 - September 2019)

*Environment: Java, Python, Javascript, R, PostgreSQL*

Served primarily as a backend developer for statistical features operating on iPathwayGuide's databases. Worked on genetic pathway analysis features for Advaita's iPathwayGuide software. Collaborated with Advaita's core development team under the guidance of their lead developer. Developed proficiency with modern software engineering principles and practices.

## Additional Projects

### [Graphics and Physics Programming with Hardware APIs](#)

Tools, games, and tech-demos developed in C/C++ primarily with OpenGL and GLSL. Examples include a CCD Inverse-Kinematic hydra demo, pinball with a bespoke collision engine, development tools for game I made for researchers, a mass-spring cloth simulation, a PBD rope simulation, and a game engine supporting features like asset and scene loading, precomputed and dynamic lightmapping, in-engine scene manipulation, and more.

### [Graphics Programming "From-Scratch"](#)

A rasterizer and a raytracer written in standard C and C++. The rasterizer renders OBJ meshes and skyboxes and the raytracer renders OBJ meshes as well as implicit spheres and planes. Each employs my personal libraries for image I/O, asset importing, and linear algebra. Accompanying these projects were a handful of implementations of image processing algorithms to test my image I/O library.

### [Game and Simulation Development](#)

Team and solo projects, primarily in Unity, also in Godot or Unreal 4. Recent highlights: a 3D painting demo employing compute shaders for image processing, two entirely hand-drawn and procedurally animated shoot-em-up games, and an unreleased FPS where I explored navmeshing with R-Tree spatial partitions and Delone triangulation. I've worked on various other gamedev projects for jobs, game jams, club activities, and personal fun.

## Leadership

Gopher Graphics Club: Workshop Officer, Video Game Development Club: Programming Officer

## Coursework

Computer Graphics, Animation in Games, Linear Algebra, Operating Systems, Machine Architecture, Data Structures and Algorithms, Automata Theory, Discrete Mathematics, Software Design and Development