RYAN DAVIS

(425) 681-0338
rdavis.dev@gmail.com
linkedin.com/in/rdavisdev
github.com/rdavisdev

Game Programmer – rdavisdev.github.io

Languages – C++, C, C#, GLSL, HLSL
Libraries – OpenGL4, STL, ImGUI, Spine C++ Runtime
Platforms – Unreal Engine 4, Unity3D
Concepts – Ray Marching, Bounding Volume Hierarchies, Shadow Mapping, Deferred Rendering
Multi-Threading, Algorithm Analysis, Design Patterns, x86 Assembly Debugging

EDUCATION

[BS] Computer Science in Real-time Interactive Simulation — **Graduating - August 2022**
DigiPen Institute of Technology

[BS] Computer Science — **Summer 2018**
University of Colorado Boulder

ACADEMIC PROJECTS

Graphics Programmer, Gameplay Programmer [UE4 / C++ / HLSL] — **In Development**
Behemyth – *Mobility Wave Defense* — *Team of 10*
- Integrated volumetric cloud shader for enriched atmosphere and dynamic weather events.
- Equipped designers with dynamic zipline mechanic for satisfying player mobility.
- Constructed waypoint and cinematic scripting tools for flexibility and ease of use.
- Established actor state queuing system allowing reliable interactions between modular components.

Graphics Programmer / Engine Programmer [C# / C++ / GLSL] — **Summer 2020**
Isles of Limbo – *2D Hack and Slash* — *Team of 11*
- Developed and optimized engine's OpenGL graphics and visual effects pipeline.
- Overhauled particle system instancing deterministic particles allowing ~10,000 particles per draw.
- Built profiling toolset for finding bottlenecks and improving engine performance.
- Implemented Spine 2D C++ runtime library to run dynamic animations.
- Applied archetype deserialization and runtime loading to decrease asset load times.

Graphics Programmer / Gameplay Programmer / Producer [C++] — **Summer 2019**
Chromatic Split – *Grid-based Multitasking Puzzle Game* — *Team of 4*
- Developed render pipeline for game's principal color mixing mechanic.
- Refined game-feel with satisfying player/camera movements.
- Implemented hierarchical tile class design for easy mechanic prototyping.
- Designed serialization system for constructing levels from external layouts.

Graphics Programmer / Engine Programmer / Producer [Java] — **Spring 2018**
Java Shooter – *2D survival shooter* — *Team of 4*
- Incorporated Android SDK's MVC library to render graphics and register input.
- Developed random level generation and enemy behavior.
- Utilized UML diagrams to plan engine design.

Graphics Programmer [C / GLSL] — **Summer 2017**
Etch – *3D Object Modeler* — *Solo*
- Implemented object construction using real-time mesh editing.
- Incorporated texture mapping for seamless surface detailing.
- Generated dynamic shadow maps to light created objects.

INDEPENDENT PROJECTS

Gameplay Programmer [Unity3D / C#] — **2015 – 2019**
Unity3D Tech Demos – *Engine Exploration* — *Solo*
- First Person Hack and Slash fighting 100+ Enemy AI
- VR Tower Defense Game mixed with Tabletop Board Game Interface
- VR Zero-Gravity with Geometry Grip Locomotion System