

CHRIS ZHONG | Game Programmer, Game Engineer, Game Developer

Redmond (Open to Relocate) | Phone: (818) 747 8878 | Email: chrisyuanzhong@outlook.com

GitHub: <https://github.com/ChrisYuanZhong> | Portfolio: <https://www.chrisyuanzhong.com>



SKILLS

Languages: C#, C++, C, SQL, HTML, Lua

Engines & Tools: Unity, Unreal Engine (C++/Blueprint), Cocos2d-x, Git, Perforce

Technical: Gameplay Programming, Performance Optimization, Stability, CPU/Memory Profiling, Debugging, Mobile Development (Android/iOS), Animation Systems, Behavior Trees, Tooling/Pipelines, Native Plugins, VR, Game Physics, Collision Detection

EXPERIENCE

KOOAPPS (UNITY, C#, ANDROID, IOS)

Seattle, WA

Game Developer (Optimization) – Snake.io

07/2024 - Present

- Reduced Android ANR rate from 0.55% to 0.40% by implementing **Crash and ANR** logging through native plugins and driving targeted fixes, which directly resulted in the app being featured on Google Play.
- Improved FPS and reduced stutter in a live Unity mobile title by using **CPU and memory profilers** to identify bottlenecks and deliver logic, architecture, dynamic asset loading, and memory leak optimizations.
- Resolved critical, revenue-impacting issues across gameplay systems, peer-written code, internal features, and third-party SDK integrations, providing fast debugging and production support on Android and iOS.

THE GAPP LAB (UNITY, C#, VR, SQL, ANDROID)

Salt Lake City, UT

Game Programmer

08/2023 - 05/2024

- Refactored a poorly structured Unity **Android VR** serious game for **Meta Quest**, turning a non-playable build into a functional medical training simulation for urinary catheter insertion.
- Rebuilt core gameplay mechanics and supporting systems to stabilize the project, improve user experience, and enable continued team iteration.
- Developed a **MySQL**-based analytics tracking system for two Unity serious games focused on depression research.

PROJECTS

GAME ENGINE SYSTEM: PHYSICS STATIC LIBRARY - *Game Programmer (C++)*

11/2023

- Independently developed a modular C++ physics static library used in my [Cross-Platform Game Engine](#), and integrated it into my [Game AI Project](#) for game AI simulation, attracting multiple fellow peers to use it in their own game engines for game development.
- Designed **rigid body** and **collision systems** with collision callbacks and a three-phase broad-phase/narrow-phase pipeline using bounding sphere, **AABB**, and **SAT** detection for rotated box colliders.
- Optimized collision queries with performance-oriented data structure choices to support scalable real-time simulation behavior.

CROSS-PLATFORM GAME ENGINE - *Game Programmer (C++, D3D, OPENGL)*

08/2023 - 11/2023

- Developed a custom C++ game engine supporting both **D3D** and **OpenGL** and features an intuitive Unity-style scripting workflow. Then built a [Side-Scrolling Platformer](#) game using the engine and my [Physics System](#) static library.
- Applied **Reference Counting** to eliminate memory leaks and improve runtime stability.
- Built a **Maya export pipeline** to export 3D models into a customized human-readable **Lua** file format to enhance readability and editing convenience, and convert it into a custom **binary** format during build time, reducing asset size by over **75%** and improved mesh loading speed by than **120x+** times.

ACT Combat with Enemy AI - *Game Programmer (UNREAL) (3RD PERSON)*

03/2024 - 04/2024

- Built modular combat, weapon collision, and state management systems using decoupled components and Blueprint Interfaces to support maintainable gameplay code.
- Implemented **enemy AI** with **Behavior Trees** and **AI Perception**, plus animation-driven combat features including combo attacks, hit reactions, weapon draw/sheath, and directional dodge invulnerability frames.

GAME AI Simulation - *Game Programmer (C++, OPENFRAMEWORKS)*

02/2024 - 04/2024

- AI Fundamental Movement Behaviors:** Integrated my [Physics System](#) to simulate physics for implementing [Craig's algorithms](#) to simulate fundamental **game AI movement behaviors** including Seek, Arrive, Flee, Pursue, Evade, Wander, and Flocking.
- Pathfinding:** Implemented the **Dijkstra** and **A*** pathfinding algorithms and evaluated their performances. Integrated **A*** with Boid, enabling them to utilize the Seek behavior from the fundamental movement behaviors to navigate towards where mouse clicked.

ALT CTRL GAME: OVERFLY - *Lead Game Programmer (UNREAL, C, HARDWARE) (10 PPL)*

01/2023 - 04/2023

- Physics Based Movement:** Independently designed and implemented an **advanced movement system** incorporating real-time **physics calculations** to realize diverse movement modes that served as the foundation of the ALT CTRL game.
- Hardware Design & Implementation:** Programmed **C code** onto **Arduino** motherboards for ultrasonic sensors as the key component of our ALT controllers. Kept optimizing hardware communications to achieve **responsive control** of the in-game balloon inflations.
- Innovative Inputs:** Designed and implemented two kinds of **input methods** for game menus and gameplay on bike pumps: scrolling and confirmation. Confirmation is achieved by “bursting”, which is to quickly pump twice, also used for “jumping” in gameplay.
- Object Inheritance Hierarchy:** Designed **inheritance hierarchy** for obstacles to make it very easy for artists to make new assets.

EDUCATION

UNIVERSITY OF UTAH - MASTER OF ENTERTAINMENT ARTS & ENGINEERING

08/2022 - 05/2024

- Relevant Courses:** C++ Game Programming, Shader Development, Advanced Game Studio, Rapid Prototyping

CHONGQING UNIVERSITY OF TECHNOLOGY - BACHELOR OF COMPUTER SCIENCE

09/2018 - 06/2022

- Relevant Courses:** Linear Algebra, Programming Language, Computer Graphics, Data Structure, Algorithm, Database

- Scholarship:** First-Class Scholarship for 2020-2021 Academic Year of Liangjiang International College

10/2021