Shuang (Richard) Li

Address: U536 15-27 Wreckyn Street, North Melbourne VIC 3051
Email: richardli65536@gmail.com
Phone: +61 482 537 101

GitHub: https://github.com/chomosuke
LinkedIn: https://www.linkedin.com/in/shuang-li-bba020181

Work Experience

Associate Software Engineer at Detector Inspector

Dec 2021 - Jun 2022
- Worked in an agile environment.
- Worked with PHP and JavaScript.
- Led the adoption of a sharable ESLint config.
- Pushed for the adoption of Data Catalog which would allowed the team to document around 500 SQL tables.

Skills

Programming Languages

Java, Kotlin, C, C++, Python, C#, Typescript, JavaScript, Dart, Rust, PHP, Haskell, Prolog, HLSL, GLSL, Golang, Matlab, Lisp(Clojure), Lua.

Platforms / Frameworks

MongoDB, Node.js, Express, React, CakePHP, SQL(MySQL, PostgreSQL), Prisma, Gin, Flutter, Android, Unity, OpenGL, Rocket.rs, AWS, Docker.

Qualifications

Bachelor of Science (Computing and Software Systems)

University of Melbourne, Jul 2019 - Nov 2021, WAM: 84.67%

Programming Related Achievements

General committee of University of Melbourne Competitive Programming Club

- Elected for year 2020 / 2021.
- Hosted two C++ workshop which aims to help our club members who have never seen C++ before to be proficient enough to use it for competitive programming.
- Overall my workshop was received positively among our club members, few of them telling me it was very useful, and they learned a lot from it.
A recording of one of the workshop is available at https://youtu.be/tM_KfYcfS4M

Wrote a Ray Tracing shader in one day using HLSL in Unity

- A realistic water shader with reflection and refraction.
- Made as a part for a university group assignment. We were required to write a Unity project generating a mountainous landscape using the diamond square algorithm, and some fairly trivial waters in that landscape.
- As a challenge to myself, I decided go above and beyond to implement transparent water with reflection and refraction using non-recursive ray tracing.
- After some nights of planning and researching, I wrote and debugged the entire shader in 1 single day.
- In the end, after some application specific optimization, I was able to lower the time complexity for collision detection of a single ray from $O(n)$ to $O(sqrt(n))$ where n is the number of triangle in our landscape mesh. I ended up achieving 80fps on my GTX 1050 ti with 32768 triangles in the landscape mesh.
Screenshot of it in action: https://github.com/chomosuke/RayTracing-in-Unity/blob/master/Screenshot.png (when it runs the water would actually move and the reflection & refraction changes with it)
Source code at:
https://github.com/chomosuke/RayTracing-in-Unity/blob/master/project-1/Assets/WaveShader.shader

9th in Unimelb Competition Programming Championship

2020, 2022

3rd in New Zealand Programming Contest Tertiary Open

2020

Personal Projects

Lumpime tracker

- A speed run of me trying to understand how long does it take for me to complete a full stack project with technologies unfamiliar to me.
- Requirement defined by my friend one night before I started the project to ensure I can not plan ahead.
- I finish the project in 5 days.
- The technology I used in this project are: Go, Gin, MongoDB, Flutter. Before this project I've only worked with flutter once on another small leisure project: https://github.com/chomosuke/catballchard. I've never worked with go or gin before this project.
- Hosted at: https://lumpime-tracker.herokuapp.com/
- Source code at: https://github.com/chomosuke/film-list

Press turn Battle System

- A plug-in that modify the game mechanics and graphic of the game engine – RPG Maker MV.
- Made upon request from one of my high school friend.
- Involves reading the undocumented, barely commented source code of the game engine written in JavaScript and overriding the appropriate functions to make the desired changes in game mechanics and game graphics.
- Me and my friend collaborated closely to establish the exact requirement for the plug-in.
- For now, I've completed the coding part of the plug-in. I am waiting for my friend to produce a sample project, after which we'll sell our plug-in on itch.io for about $15.
Source code available upon request.

Project Rocket

- A very simple time-killer game involving a rocket flying through space.
- Made in my spare time in high school for fun.
- The entire game was made using only Open GL which I learned from tutorials on Android's official developer guide, as at the time I didn't know game engine existed.
- I learned most of my programming skills including many programming practices, principles and OOP design pattern during the making of this game.
- The game is published on Google Play which has so far gained more than 1000 downloads.
Google Play link: https://play.google.com/store/apps/details?id=com.chomusukestudio.projectrocketc
Source code at: https://github.com/chomosuke/ProjectRocketC

PRC Android 2D game engine

- An 2D game engine for Android built upon Open GL with GLSurfaceView. - Made as a part of Project Rocket before I decoupled it and made it a standalone game engine. - It can draw textures and various geometric shapes, do collision detection etc. Source code at: https://github.com/chomosuke/PRCAndroid2DGameEngine

Other Achievements

Duke of Edinburgh's Bronze Award

2019

Successful participant in IMMC

2018

NZQA Outstanding Scholar award

2017 A award given to the 50 most outstanding NZ high school students. I won it as a year 12 against all the year 13 students.

4th & 6th in senior Auckland Math Olympiad

2017 & 2018

1st & 3rd in Auckland math Olympiad

2017 & 2018