MING CHENG

chengmingjtt123@gmail.com
mingchengzack.github.io
6197817840
Unit 320, 3979 Nobel Drive, San Diego, CA, 92122
mingchengzack

Skills

PROGRAMMING LANGUAGE

C
C++
Python
JavaScript
HTML
CSS
Golang

FRAMEWORKS

React.js
Boost Asio
OpenGL
Design Patterns

TOOLS

Git
Firebase
MongoDB
Protocol Buffers
AWS
MySQL

OS EXPERIENCE

FAT-based File System
User-Level Thread Library
Shell

Education

University of California, San Diego — Sept. 2019 to Current
Master of Science Computer Science 2021
GPA: 3.96/4.0

University of California, Davis — Sept. 2015 to June 2019
Bachelor of Science Computer Science and Engineering 2019
GPA: 3.7/4.0

Experience

XCOM Labs, Inc. — San Diego, CA
Software Engineer Intern — June 2020 to Sept. 2020

- Worked on a team to develop a service to deliver remote rendering for VR in Linux environment using **C++** and **Yocto Project.**
- Worked on profiling and benchmark on the service using **Gperftools** and **Google Benchmark**.
- Worked on optimization of the service and code such as implementing lock-free queue.

Gamification of Nutrition Literacy — University of California, Davis
Dr. Lisa M. Soederberg Miller's Team — Jan. 2019 to June 2019

- Worked in a team to develop a learning system with web-based games.
- Implemented two original games with various levels, login system for user interaction, leader board and badges rewards system in **JavaScript, HTML** and **CSS.**
- Implemented game physics, scene flow in **JavaScript** with **Phaser 3**, a framework for 2D games, designed game maps.
- Used **Firebase**, a cloud-hosted NoSQL database to collect user data. LINK

Client-Server for Warcraft II — University of California, Davis
Prof. Christopher Nitta's Multiplayer Team — Jan. 2019 to Mar. 2019

- Worked in a team to develop multiplayer support for Warcraft 2, a real-time strategy game (MOBA), on Linux, Mac OS and Windows.
- Implemented multiplayer system with Client-Server model in **C ++** with **Boost Asio**, a cross-platform C++ library for network programming.
- Implemented login system with authentication, and message system for pre-game and in-game chatting.
- Used Protocol Buffers to serialize user and game data for efficiency.

Projects

Key-Value Storage Service — July 2020 to Current

- Developed fault-tolerant, distributed key/value storage service on top of Raft using **Golang**.
- Implemented Raft library to achieve fault-tolerance using **Golang.**

MapReduce System — Mar. 2020 to Apr. 2020

- Developed a distributed and worker-fault-tolerant MapReduce system for handling reading and writing files using **Golang**.
- Utilized master-worker model and RPC for assigning tasks and parallel execution of map and reduce functions.

Sorting Visualizer — Jan. 2020 to Feb. 2020

- Developed a web-based visualization tool for various sorting algorithms using **React.js, Javascript, HTML,** and **CSS.**
- Applied Bubble Sort, Insertion Sort, Selection Sort, Cocktail Shaker Sort, Gnome Sort, Bitonic Sort, Shell Sort, Quick Sort, Merge Sort, Heap Sort, Radix Sort, and Bucket Sort for visualization.

Path Finding Visualizer — Dec. 2019 to Jan. 2020

- Developed a web-based visualization application of path-finding algorithms using **React.js, Javascript, HTML,** and **CSS.**
- Applied Dijkstra's Algorithm, A-Star Search Algorithm, Depth-First Search, Breadth-First Search, and Greedy Best-First Search for visualization.

3D/2D Drawing System — Sept. 2018 to Dec. 2018

- Developed a system that can draw and transform lines, polygons and simple polyhedral in **C++** with **OpenGL**.
- Applied DDA and Bresenham line drawing algorithms to draw lines, the scan-line algorithm for rasterizing polygons, and the Cohen–Sutherland algorithm for two-dimensional clipping.
- Applied Phong lighting model, Gouraud shading and the Painter's algorithm to display colored 3D objects.