Liam Wynn

Portland, Oregon

wynnliam@pdx.edu wynnliam.github.io 503 747 9596

Qualifications • Comfortable in a wide range of programming paradigms and languages. • Experienced writing thorough documentation on projects. • Use math and theory to derive algorithms that I implement, test, and optimize. • Adept at learning new tools and technologies. • Thrive in both team and individual settings.

Education

Portland State University Portland, Oregon
B.S., magna cum laude, Computer Science, Mathematics Minor 2014 - 2019

- GPA: 3.85
- Academic Honors: Dean's List (7 semesters).
- Inducted into Phi Kappa Phi National Honor Society 2018.

Experience

Standard Insurance Company Portland, Oregon
Software Development Intern July 2018 - November 2018

- Worked on the back end of web applications using Java and Springboot.
- Did code reviews, unit testing, and agile/scrum software development.
- Worked with several other teams to complete projects.

Portland State University, Portland Portland, Oregon
Computer Science Tutor April 2016 - August 2019

- Assisted lower division Computer Science students with their course work and development endeavors.
- Helped students with problems that ranged from systems programming to rudimentary logic and discrete mathematics.

Projects

VA Audiology Web App - *Typescript, Node.js, SQL*
January 2019 - June 2019
For my senior capstone project, I worked with a team to add a back end to an existing application for the VA. This included a SQL database, a Node.js backend, and rewriting several parts of the existing front end to utilize the latter two systems.

Raycore - *C*
April 2018 - February 2020
A game engine built around ray casting which is a technique for rendering pseudo-3D environments. Rendering uses several numerical approximations and low-level optimizations to improve runtime efficiency.

Texture Generator - *Python, Tensorflow*
November 2018 - Present
Employed a variational autoencoder to generate texture images. Both the encoder network uses a convolution layers with leaky ReLU activation. The decoder uses transposed convolution layers with leaky ReLU activation. Uses the Adam Optimizer algorithm to train.

Technical Skills

Languages: C, C++, C#, Python, Java, Javascript, Typescript, SQL, x86-64 Assembly, Haskell, Lisp, Frink, Pie
Libraries and Frameworks: STL, Tensorflow, NumPy, OpenGL/WebGL, SDL, jQuery, Node.js, Flask, Express.js, React. Angular, Asyncio, Pyro4, SCOOP
Paradigms: Object-Oriented, Functional, Imperative, Full Stack, Low-Level/Assembly, AI/ML
Computer Skills: GNU/Linux, Shell, Git Command Line, Vim, GDB, Valgrind, Docker, Latex, Google Cloud, Google AI Platform