Bao Luong

West Covina, 91791 (**Wiling to relocate*)
720-299-1002 | bnluong@uci.edu | www.linkedin.com/in/bao-luong | www.github.com/bnluong

Education

University of California, Irvine — Irvine, CA
Bachelor of Science in Computer Science — Sep 2017 - Dec 2019
GPA: 3.57/4.00

Coursework: Data Structures and Algorithms, x86 Architecture, System Design, Projects in Operating Systems, Programming Languages, Compilers and Interpreters, Databases, Information Retrieval, Artificial Intelligence, Machine Learning, Computer Networks, Software Engineering, Software Testing & QA, Computer Games Development

Skills

Proficient Languages:	Java, C++, C
Familar Languages:	C#, Python
Environments:	Windows, Linux
Tools:	Office Suite, Eclipse, Visual Studio, Git, MySQL, Unity, LaTeX

Projects

Nightmare Survivor (2019) *Unity, C#* — https://github.com/Arma15/BestGame

- Designer and programmer in a team which functioned as an indie developer
- Utilized Agile methodology during development through the use of Sprint
- Developed a prototyped 3D, first-person computer game featured standard key binding, camera control, collision detection, ray tracing, and basic AI components

Kaggle Competition (2019) *Python*

- Fit and evaluated different types of machine learning classifiers for the UCI Kaggle data
- Implemented the following learners: logistic regression, random forest, kernel SVM, boosting ensembles, and stacked ensembles
- Achieved top 11% in the competition

Dynamic Memory Allocator (2018) *C*

- Developed a functional dynamic memory allocator which functions similarly to the malloc(), free(), and realloc() system calls

Simple Search Engine (2018) *Python*

- Developed a simple search engine that performs web crawling, indexing, and retrieving over a static corpus
- Utilized OOP design and implementation. Optimized retrieving and indexing time by performing pre-checks at startup

Wumpus World AI Agent (2018) *C++*

- Developed a knowledge-based agent to solve the Wumpus World game problem in AI

Simple OS (2017) *Java*

- Developed a simple operating system with both front-end and back-end components
- The OS allows multiple users to save files to disks and print files
- The goal was to explore concurrency with multithreading with a GUI to display what happens when the system is running