

Aram Kazorian

aramk@berkeley.edu (818) 808-9029

Education

University of California, Berkeley

Class of 2023

B.A. in Computer Science

Coursework:

- Computer Graphics (Self-taught, In Progress)
- Data Structures
- Algorithms (In Progress)
- Artificial Intelligence
- Database Systems (In Progress)
- Machine Structures
- Circuits and Linear Algebra I and II
- Discrete Math and Probability

Experience

UC Berkeley - Undergraduate Student Instructor

January 2021 - Present

Data Structures and Algorithms

- Led daily Discussion and Lab sections, where students would implement Data Structures and Algorithms taught in the course.
- Held Office Hours to individually help students debug their projects, labs and other coursework.
- Reinforced student knowledge of Java programming methodology, data structures, analyzing asymptotic runtime, along with sorting and graph algorithms.
- QA'd and helped revamp course projects, labs, among other course related content.

UC Berkeley - Computer Science Mentors (CSM)

August 2020 - December 2020

Junior Computer Science Mentor - Data Structures and Algorithms

- Taught and mentored a group of 4-5 students on the topics of Data Structures and Algorithms students by using problem based worksheets.
- Wrote mini-lectures and created collaborative environment to work on problems covering course topics.

UC Berkeley - Academic Intern

January 2020 - May 2020

Structure and Interpretation of Computer Programs

- Assisted course staff in weekly lab sessions by QA'ing students to establish a foundation of programming and the interpretation of programs.
- Course content taught includes Python syntax, Object-Oriented Programming, recursive data structures, and the internals of a language interpreter.

Projects

Rasterizer

UC Berkeley, Computer Graphics

- Implemented a simple rasterizer using OpenGL, that included features such as drawing triangles, hierarchical transforms, and texture mapping.
- Images and textures can be rasterized using simple point sampling or using antialiasing via supersampling.
- Texture sampling is implemented using bilinear interpolation, while Level Sampling is implemented with the use of barycentric coordinates, among other calculations.

Gitlet

UC Berkeley, Data Structures

- Utilized knowledge of file systems, algorithms, and data structures to implement an efficient and usable version control system similar to Git.
- The API mimics Git's basic local commands such as init and add, and allows for committing, branching, merging, and status checks.

Lines of Action

UC Berkeley, Data Structures

- Implemented an API that includes a GUI following the rules of the game.
- The API allows for a full two-player game and includes a fully functioning AI, which calculates moves based on the A* search algorithm using a heuristic designed for the problem space of the game.

Technical Skills

OpenGL, Java, Python, C, C++, Git, UNIX