

Joey Zhu

☎ (+1) 925-858-2048 | ✉ joey.j.zhu@berkeley.edu | 🏠 <https://joey-j-zhu.github.io/> | 📷 joey.j.zhu | 📺 joey-j-zhu

Education

University of California, Berkeley

B.A. COMPUTER SCIENCE AND PHYSICS

Berkeley, CA

Aug. 2019 - Exp. May. 2023

- Data Structures, Efficient Algorithms, Random Processes, Linear Algebra and Differential Equations, Information Systems, Computer Architecture, Machine Learning
- Tensor Algebra, Analytic Mechanics, Perturbation Theory, Statistical Physics, Quantum Mechanics, Classical Electrodynamics

Skills

Programming Python, C, RISC-V, Java, JavaScript, HTML, CSS

Tools Git, Numpy, Scipy, Tkinter, OpenCV, OpenMP, Photoshop, Illustrator

Soft Skills Planning, Organization, Problem Solving, Graphic Design

Work Experience

UC Berkeley EECS Dept.

ACADEMIC INTERN

Berkeley, CA

Jun. 2020 - Oct. 2020

- Worked with instructors in Data Structures (CS 61BL) and Discrete Mathematics (CS 70) courses to debug staff auto-graders and student labs
- Led sections of two groups of 25 students each, created practice material to help build intuition for data structures and mathematical proofs

UC Berkeley IEEE

MARKETING COMMITTEE, GRAPHIC DESIGNER

Berkeley, CA

Aug. 2020 - Jul. 2021

- Created main designs for yearly apparel order
- Led apparel design project with two other designers and coordinated distribution logistics with executive members

Projects

Sorting Hat

DIVIDE A SET OF ITEMS INTO SEVERAL GROUPS TO MAXIMIZE A TOTAL SCORE, MAINTAINING A COST THRESHOLD

Berkeley, CA

Dec. 2020

- Proposed and led implementation of the main solution structure, combining gradient descent and genetic algorithms
- Improved quadratic-time validity computations to linear-time, and linear-time sampling to amortized constant-time
- Generated pathological inputs to test the resilience of solver

NumC

ACCELERATE NATIVE C/PYTHON ARRAY-BASED ROUTINES, WITH FOCUS ON L1 CACHE AND INSTRUCTION SET OPTIMIZATION

Home

Jul. 2021

- Used a combination of Intel x86 intrinsic vectors, OpenMP threading, cache optimization, and minimizing instruction set sizes
- Added transposition to matrix multiplication routine for more cache hits on successive dot products, speedup from factor of 5 to 80
- Minimized memory allocations, accesses, and compulsory cache misses, pushing matrix powering speedup from 780 to 820

Perlin Mapping

TUNE PERLIN FRACTAL NOISE PARAMETERS TO VISUALLY APPROXIMATE AND MANIPULATE IMAGES

Home

Jun. 2021

- Lowered squared raster error from 300k to less than 1k for images with varying features, via gradient descent and error-passing for successive octaves
- Adjusting program and exploring more parameters for gradient descent process to behave more predictably and produce numerically similar parameters for visually similar images or successive frames