Joseph Tan

U.S. Citizen | (626) 693 6386 | **Email: dghosef@stanford.edu**
Github: github.com/dghosef | **Linkedin: linkedin.com/in/dghosef** | **Website: dghosef.me**

Proficient in C, C++, Python; **Familiar with** ARM Assembly, Lisp

Education

Stanford B.S. Computer Science *2020-2023(expected)*

Current Cumulative GPA - 3.95/4.0

Relevant Coursework

CS106B Programming Abstractions(A-): *C++, Recursion, Data Structures, Object Orientated Programming*

CS103 Mathematical Foundations of Computing(A): *Discrete Mathematics, Formal Language Theory, Finite Automata, Regex, Complexity Theory*

CS107E Computer Systems from the Ground Up(A): *I/O, ARM Assembly, Bare Metal C, Driver Development, Interrupts*

CS143 Compilers(in progress): *Compiler Implementation, Lexing, Parsing, Semantic Analysis, Intermediate Representations, Code Generation*

CS111 Operating Systems(in progress): *Concurrency, Synchronization, Scheduling, Processes, Virtual Memory, File Systems*

Experience

Research Assistant

UCLA Micro and Nano Manufacturing Lab *(June 2017 - August 2017)*

- Built high-resolution DXF to PDF file converter for circuit manufacturing website
- Helped create interface between laptop and boat speedometer for lab experiments

Robotics Team Founder

Founded high school VEX robotics team

- Taught STEM topics(basic programming, practical physics, etc) to high school students
- Started team, mentored students, developed website, raised funding

Technical Projects

Berryboy*(C)*

Gameboy emulator running in a baremetal environment

- Executes Gameboy ROMs by emulating the Gameboy CPU, GPU, Interrupt mechanism, and I/O
- Runs in baremetal(OS-free) Raspberry Pi environment with custom controller driver, use of GPIO interrupts, and manual framebuffer manipulation

Comper*(C++)*

Jazz backing track generator from chord progression and context-free-grammar based style file

- Generates a drum track, a walking bassline, and piano chord playing with jazz voicings
- Implemented grammar parser, music generation logic, midi file generation, and music-specific data structures

FPL Team Generator*(Python)*

Fantasy Premier League(FPL) team selection algorithm(**dghosef.me/fpl-writeup**)

- Scraped soccer player statistics from the FPL API and saved in Pandas dataframe
- Developed algorithm to predict future player performances based on past performances, upcoming fixture difficulty, etc
- Utilized linear programming solver to maximize predicted future performance levels under constraints of FPL rules to build team