**William Alex Greenwald**
walexgreen@gmail.com | (619) 371-2170 | william-greenwald.com

## Education

**University of California - Santa Barbara** Sept 2021 - June 2025
**Bachelor of Science, Computer Science**
- **Most relevant CS Courses:** Object Oriented Design; Comp. Organization & Architecture; Data Structures & Algorithms; Digital Design; Operating Systems; Comp. Networking; Automata & Formal Languages
- **Math Courses:** Multivar. Calc, Linear Algebra & Diff. Equations, Discrete Mathematics, Prob. & Stats

## Skills

**Languages: C, C++, MIPS Assembly**, HTML, CSS, Verilog, Python, Java
**Technologies: Github, VS Code**, Numpy, Sockets, Makefile, SCons, LaTeX, pyRTL, Unix, T32
**Areas: Operating Systems**, TCP/IP Networking, Computer Architecture/Design, Data Structures
**Other: Agile (Scrum/Kanban)**, Native Bilingual (Spanish+English), Guitar, Piano, Club Ultimate Frisbee

## Work Experience

**Qualcomm** | Embedded System Software Engineer Intern June 2023 - Sept 2023
- Worked with the Multiprocessor Communications team to implement the usage of an open source IP stack to communicate between 2 processors within a Qualcomm proprietary system on chip (SOC).
- Created a custom physical transport layer for the IP stack by leveraging APIs developed by other teams.
- Wrote BSD socket & raw API tests, which were cross compiled for a Qualcomm platform; Traced through the executable/C code using T32 to debug runtime errors, memory issues, and interrupt system.
- Presented my work to other interns, managers, and engineers at the end of internship.

**Seaworld San Diego** | Ride Operator June 2022 - Sept 2022
- Seasonal summer ride operator for “Shipwreck Rapids” ride at Seaworld San Diego.
- Loaded, unloaded, grouped, and monitored ~500 guests/hour at this attraction.
- Used teamwork, communication, and efficiency to maintain guest and employee safety at all times.

## Projects

**System Call & Pipe Integrated Operating System** C programming language
- Developed an operating system in C that enabled functionality of traps & interrupts.
- Implemented the use of basic system calls and pipes, which enable reliable inter-process communication.
- Gained an understanding of how operating systems deal with things like threading, time sharing, multiprogramming, and memory management when handling multiple processes.

**Array Encoder/Decoder & Python instruction assembler** MIPS Assembly, Python
- Assembly program, takes an array of integers, calls an encode and decode ‘function’, which takes the values and applies/undoes arithmetic (power, subtraction) to them and returns/prints values.
- Python program that takes in a MIPS Asm. instruction and displays binary & hex of instruction on tkinter.
- Gained assembler knowledge with python assembler; worked with jumps, the stack, & branches in Asm.

**Binary search tree card game** C++, Python
- Created a card game by implementing a binary search tree class from scratch in C++
- Used .txt files as inputs. Learned how to use ifstream to read and parse through the .txt file contents.
- Mapped unique strings from .txt files to weighted int values, and translated those ints back to the same strings at the end of program for more efficiency.
- Average runtime was ~2.6 times faster vs. basic linear search comparisons in personal tests.
- Inserted, compared, and deleted all values as nodes in 2 unique BSTs. Program represents a game simulation, where 2 .txt files of strings representing cards are compared using my BST.
- Wrote python script to create random test cases. Used numpy for consistent tests written onto .txt files.