Jim Palomo

Computer Engineering Major · The University of Illinois at Chicago

847-345-2180 | jimppalomo@gmail.com | jimpalomo.github.io | JimPalomo | Jim-Palomo

Education

University of Illinois at Chicago — *Chicago IL*

Bachelor of Science in Computer Engineering — *Aug 2018 - May 2022*

- Cumulative GPA: 3.76
- Undergraduate Coursework: Data Structures; Introduction to Embedded Systems; Introduction to Logic Design, Mathematical Foundations of Computing, Introduction to Differential Equations, Calculus III, Applied Linear Algebra.

Experience

Research Internship for Electrical & Computer Engineering — *Chicago, IL*

Intern, Co-Op Aide — *Jun. 2019 - Aug. 2019*

- Worked on an open-source simulation platform for computer system architecture called gem5
- Established connections among different CPU chip-sets such as ARM & x86 with memory controllers, caches, and interconnects
- Cooperated with an engineering professor and a Ph.D. student
- Gained knowledge on Object-Oriented Programming for Python and C++

Projects

Back-End Navigation — *UIC*

C++ · XML · Valgrind · Linux · GNU Make — *Apr - May 2020*

- Designed an application that allows the user to observe the back-end functions of loading a map, building a graph, and finding the shortest path between two separate locations.
- Integrated Dijstrakia's algorithm to find the shortest path among two points.
- Implemented open-source maps from openstreetmap.org of UIC's East Campus.
- Debugged using CLion & VSCode, (data structures) Map, Graph, Stack, Vector, Set. Queue.

DIVVY Data Hashing — *UIC*

C++ · CSV · Valgrind · GNU Make — *Apr 2020*

- Developed an application that hashes station and trip data from DIVVY bike-sharing company.
- Created a hashmap with separate hash functions for over 1500 trips and 580 bike IDs data.
- Added multiple commands: search by station id, abbreviation, trip id, bike id, nearby stations, and similar trips
- Debugged using CLion & VSCode, (data structures) Vector, Hashmap.

Threaded AVL Tree — *UIC*

C++ · Catch Framework (unit testing) · Valgrind · Linux — *Mar. 2020*

- Created a general-purpose threaded AVL tree class that dynamically grows that contains insert, copy/construct, print tree, height, rotations, and search keys among a specified range.
- Designed with the notation of each node contains a key, value, left/right pointers, boolean for threading, and height.
- Debugged using CLion & VSCode, Valgrind (memory leaks), (data Structures) AVL tree, vector, stack.

Amazon Autonomous Bot — *Self*

Python · Google APIs · Google Sheets — *Jul 2019*

- Programmed a web scraping bot that gathers product information on Amazon: price, URL, name
- Updated spreadsheet with data from initial scrape which sends an email notification to the user

Raspberry Pi Smart Monitor — *Self*

Linux · JSON · Python — *May 2019*

- Developed through Raspbian OS using open-source code via GitHub (MichMich).
- Implemented personal calendar, cryptocurrency stock tracker, weather API, date & time, etc.

Skills

Languages C · C++ · Python · ARM Assembly

Software/Tools Linux · Git · Shell Scripting · SSH · Catch Framework · Valgrind