James A. Zeuch

9424 N. Military Ave, Oklahoma City, OK 73114
(405)-905-9140
zeuchtony@gmail.com

Education

Bachelor of Science in Computer Engineering December 2018
Oklahoma State University GPA: 3.66

Technical Skills

Programming Languages and Tools:
Strong - Java, Python
Proficient - C, C++, Matlab, Verilog
Basic - C#, XML, HTML, Javascript, MySQL
Learning – Django, Web Design, Databases
Unique Library Experience - FFMPEG, Google Test
Framework Exprience – Visual Studio, Atlassian Suite, Linux, Git

Other Experience: VLSI Design, Circuit Analysis, Computer Hardware, Agile

Soft Skills: Strong Work Ethic, Effective Problem Solver, Adaptable Worker, Passionate Learner, Strong Diagnostic Skills, Strong Communication Skills, Collaborative

Work Experience

Northrop Grumman
Intern – Software June 2017 – Dec. 2018
Associate Software Engineer Jan. 2019 – Apr. 2020
Software Engineer Apr. 2020 – Current
- Create and patch programs written in C/C++.
- Create programs in python to automate and/or simplify work.
- Test program functionality using Google Test.

Oklahoma State University
Community Mentor January 2017 – May 2018
- Become acquainted with many residents and help them when necessary.
- Maintain flexibility to be available to assist other student staff members.
- Enforce university policy.

The 3 Geeks
Intern – Bench Technician October 2013 – June 2014
- Diagnose problems with computers and examine possible solutions.
- Replaced computer hardware components.
- Remove malware, spyware, etc. and install missing software(drivers, updates, etc.).
- Required working on 1-3 computers at a time.

Relevant Projects

Puzzle Solver and Player Website (in-progress)

Description: Website using Django made to solve puzzles like sudoku, kakuro, and kenken. Website portion and multiple solvers are still in progress as well as the website. Plan to expand it to actually allow you to play some puzzles yourself, maybe including ones that are created by program.
-Created the soduku solver in python that has been proven to work for even the hardest sudoku puzzle in a relatively short time(less than 10 seconds).
-Working on website design and adding the actual solver to it using Django.
-Working on additional solvers and trying to expand into realm of allowing one to play the puzzles.
-May expand on showing logical steps to solve the puzzles because it seems like an interesting idea to gain more valuable experience.

Bit Flipping Tool (Apr. 2020 – May 2020)

Description: Python program that took a folder as input and rewrote any necessary structs in the C, C++, and header files to switch the order of the bits. This would allow the program to work on a machine and operating system that uses a different endian structure.
-Implemented a structure to handle bit flipping shorts, ints, and longs by using #ifdefs to allow it to continue working on current platform.
-Copies file structure to prevent destroying code should something unexpected happen.
-Saved weeks of rework and was broad enough to be applied to other systems that needed it.

Mercury Robotics Robot (Aug. 2018 – Dec. 2018)

Description: Robot that could complete an obstacle course, as well as an autonomous section of the course. Robot had to also be able to be controlled over the internet.
-Helped with the server side python program that allowed the robot to receive and interpret commands.
-Implemented the client side python program that ran using the keyboard as a controller using pickle before sending the instructions to make sending fast.

Rogowski Coil Ammeter (Jan. 2018 – May 2018)

Description: Device that could read the amount of amperage going through a wire down 100 micro amps. It sent a graph of the amperage over the internet to a graph on a webpage to display.
-Helped with the javascript on the webpage involving plotly in order to display our data.
-Created all our diagrams in as much details as was feasible for our project.

Minesweeper (Oct. 2015)

Description: Java program to simulate the classic game of minesweeper.
-Implemented model class to keep track of basic data and build game's mechanics.
-Implemented controller class to design the board in paint and handle the player's moves.
-Implemented view class to display the board to the player.
-Added a feature for revealing all surrounding spaces if no mines exist adjacent to clicked spot.
-Later added support for different difficulty levels.
-Later added a file menu, which included a save feature, load feature, new game, and quit.