Matt Bommer

mbommer@utexas.edu
(409) 363-9581

EDUCATION:

University of Texas, Austin, TX – Computer Science Major within College of Natural Science
Bachelor of Science in Computer Science – Expected Graduation December 2020
Overall GPA: 3.63
National Society of Collegiate Scholars, Spring 2017-Present

EXPERIENCE:

Dynamic Graphics Inc. - Junior software tester; Alameda, CA June 2019 - August 2019
- Verified the consistency of geological visual structures in DGI's CoViz software using SmartBear's GUI testing software (Test Complete).
- Developed a design pattern for building and replicating fully automated GUI tests.
- Improved test harnesses and testing methods alongside Senior Software testers by fixing recurring bugs and updating testing procedures.

Chipotle Mexican Grill – Cashier; Austin, TX May 2018 – September 2018
- Provided excellent customer service to create a positive guest experience

Beaumont Municipal Tennis Center – Instructor; Beaumont, TX March 2015 – April 2016
- Coached and mentored youth members to help them improve as tennis players

PROJECTS:

Seinfeld Box (In-Progress)
* A glorified door sensor that plays the Seinfeld Baseline upon a doors opening. The project utilizes weak magnetic fields (created by ordinary magnets) and a Raspberry Pi equip with a magnetic sensor to determine if a door is ajar or not. If the door becomes ajar, then the Seinfeld Baseline begins to play.

Mini Shell (Fall 2019)
* Program is a small implementation of a true bash shell. Shell takes in an executable and gives user the option of running said executable in the foreground or the background (shell still operable while the process runs). Also allows for processes to be moved between the foreground and background.

Pascal Compiler (Fall 2019)
* Basic compiler for the Pascal programming language. It's separated into three main phases: Lexical analysis, Parser and Code generator. Upon Lexical analysis, code is tokenized and sorted into its specific classifications (Reserved word, Identifier, Delimiter, etc). Following analysis, code is then parsed into an intermediate tree form and finally written as binary in the code generation phase.

COURSEWORK:

Discrete Math; Data Structures; Linear Algebra; Python; Computer Architecture; Operating Systems; Compilers; Symbolic Programming

TECHNICAL SKILLS:

Proficient with: Java, Python, Microsoft Office, Eclipse, Linux, Visual Studio
Familiar with: C, HTML, Assembly, Clojure, Git, SmartBear's TestComplete, Bash

ACTIVITIES:

Absolute Texxas Member, Fundraising for St. Jude Children's Hospital 2018 - Present
Southeast Texas Humane Society, Volunteered 80+ helping animals find homes 2013 - 2017