Category Two Narrative

Briefly describe the artifact. What is it? When was it created?

This artifact is a new creation but is loosely based on a project that I did in CS-300, my data structures and algorithms course in the fall of 2023. That course had us create an app that loaded data from a spreadsheet and then allowed the user to either perform quick sort or selection sort on the data. This new artifact is an applet that allows users to choose between bubble sort, selection sort, quick sort, and merge sort, select a number of values, and then watch the data be sorted in a bar graph animation.

Justify the inclusion of the artifact in your ePortfolio.

This artifact solidly demonstrates my ability to implement various sorting algorithms and showcase their efficiencies. It’s clear when using the application that different algorithms have different times, and this tool enables users who may not have a technical background to understand these different algorithms. This artifact is pretty clearly intended to prove my accomplishment of the course objective relating to algorithms because I didn’t feel that my improvements to my project for categories one and three showcased that ability. People in all industries are looking to data visualization to draw conclusions about how best to proceed in business, and this project showcases my ability to be a part of this emerging trend in computer science.

Did you meet the course objectives you planned to meet with this enhancement in Module One?

I did meet the objective that I planned to meet:

Design and evaluate computing solutions that solve a given problem using algorithmic principles and computer science practices and standards appropriate to its solution, while managing the trade-offs involved in design choices.

This artifact is a quite literal interpretation of this objective; its function is to demonstrate the efficiencies and mechanisms of these various sorting algorithms along with the help of visualization. Therefore, it completes this goal of the course. In addition to this objective I set out to meet in module one, I also demonstrated the following objectives:

Employ strategies for building collaborative environments that enable diverse audiences to support organizational decision making in the field of computer science

Demonstrate an ability to use well-founded and innovative techniques, skills, and tools in computing practices for the purpose of implementing computer solutions that deliver value and accomplish industry-specific goals

These two objectives are evidenced by the fact that this tool takes a very technical concept and creates an output that can be understood and used by almost anyone regardless of experience with computer science. One of my great strengths is bridging the gap between audiences with differing levels of knowledge, and I think this tool is a perfect liaison between diverse audiences from developers who will understand and appreciate the tool's visualization and non-developers who will come to understand how computers sort data by seeing this animation.

Reflect on the process of enhancing and modifying the artifact.

This process really helped me get acquainted with the pyplot package and all of the capabilities of visualizing data with Python. Data analysis is such an important world in the burgeoning world of big data, and the ability to create dashboards and other visual representations of data allows non-technical audiences to connect with the content. The process involved a lot of debugging as the sorting algorithms can be finicky. There are a couple of different implementations that are popular for each one, and with the added task of validation, there are a lot of moving parts, and you must make sure everything is incredibly precise. I started by just coding the sorting algorithms without the visualization aspect. I then found logical places to include plotting functions so that the animation was comprehensive and coherent. This was particularly difficult for the merge sort and quick sort algorithms as I implemented them both recursively. I then came up with a method to create a color gradient proportional to each value to add to the accessibility of the digestibility of the animation. The final component was implementing the GUI using the PySimpleGUI library. I learned that it is much harder to code application logic and then implement an interface than doing both simultaneously. The logic of the sorting algorithms and their visualization running in a console app actually wasn't terribly difficult. Validating input and accounting for all of the callback functions of the GUI elements required a lot of time testing and debugging. The PySimpleGUI library was also brand new to me, so I spent a lot of time working through that documentation to get the interface I wanted with the required functionality. It was ultimately a big learning experience in the domain of creating applets and using online resources to learn how to effectively use a new library.