Ali Alshara

11/30/24

CS 499

Milestone 3

The artifact I selected for my ePortfolio is a custom implementation of a sorting algorithm, specifically designed to sort numeric datasets efficiently while demonstrating foundational skills in algorithms and data structures. This project will showcases a step by step implementation of sorting logic using iterative and recursive approaches. The artifact includes functionality for sorting integers, floats, and other data types while maintaining flexibility through modular design. It also provides insight into how trade\ offs in algorithm design impact performance and complexity.

I chose this artifact because it demonstrates my ability to implement and optimize fundamental algorithms while effectively leveraging data structures. Specifically, the sorting algorithm highlights my skills in designing efficient logic, applying principles such as divide-and-conquer (if using merge sort or quicksort), and managing memory usage in iterative and recursive functions. As part of my enhancements, I added error handling to manage invalid input, improved comments and documentation for clarity, and introduced a comparison counter to measure algorithm performance. These improvements reflect my ability to refine solutions while maintaining a focus on performance and usability.

This artifact aligns closely with Outcome 3, which focuses on designing and evaluating computing solutions using algorithmic principles while managing trade-offs. For example, I explored the performance of different sorting techniques and optimized the solution to minimize time complexity while ensuring accuracy. Additionally, this artifact demonstrates my growing understanding of Outcome 4, as I utilized techniques and tools to implement a solution that delivers measurable value. By enhancing the artifact, I demonstrated proficiency in evaluating and improving algorithms to ensure reliability and efficiency.

The process of enhancing and modifying this artifact provided several learning opportunities. For instance, while adding a comparison counter to track algorithm performance, I gained a deeper understanding of time complexity and its impact on sorting large datasets. Refactoring the code to include error handling and modular functions highlighted the importance of writing maintainable and reusable code. A significant challenge was balancing optimization with readability, as making the algorithm faster sometimes introduced complexity. Through this process, I learned how to evaluate and manage tradeoffs to deliver a robust solution.

Moving forward, I plan to explore additional enhancements for this artifact, such as implementing alternative sorting algorithms. I also aim to add support for sorting user-defined data types, further demonstrating my ability to work with diverse data structures. These enhancements will help me progress toward Outcome 5, as I integrate security measures like input validation and address potential vulnerabilities in sorting logic. This artifact will remain a strong representation of my skills in algorithms and data structures, showcasing my ability to analyze, design, and improve computational solutions.