Milestone Three – Algorithms and data structures

Christian Bass

SNHU

Artifact:

I have two artifacts for this project, the first is my final project from “CS-260: Data Structures and Algorithms”. The goal of this project was to complete a binary search algorithm that could import data from a CSV file using a course provided CSV parser, create a binary search tree, and search the tree for a data entry. The second artifact is from “IT 145: Foundation in Application Development”. It is a basic stock system that accepts input from the user on how many apples and oranges they have, how many they should have total, and then does the calculation for how many they need to buy. I am also using this artifact for the databases portion of my project.

I chose these artifacts because I feel they are a good fit for demonstrating my skills working with algorithms and can be improved in a variety of ways. The CS-260 project will allow me to finish an algorithm I failed to create in the past while implementing an additional one and making other improvements to the overall code. The IT-145 project is a simple base that will allow me to add a new algorithm that will pair will with my databases project.

I improved the CS-260 artifact by implementing a new linear search algorithm and completing the original binary tree search algorithm. Being able to run each algorithm in the same environment allows the user to clearly see the speed difference between binary and linear searching. I also did other various small improvements such as improved commenting and more error checking. For the IT:145 artifact I added a hashing algorithm and password functionality to secure the database.

Course Outcomes:

I did meet the four course outcomes I planned to in my initial enhancement plan. My artifact met outcomes one, three, four, and five. I did this by ensuring that my code followed industry best practices, was commented, and used object oriented design for easy maintainability. I also demonstrated my ability to implement various search algorithms and understand complex code and logic problems.

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

I met this outcome by creating clear readable code that complies with C++ standards and has extensive commenting explaining all code functionality.

**Outcome Three: 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**

To meet this outcome I worked through logic problems to produce fully functional code complete with three different types of algorithms. These algorithms are bug free and well suited to the intended use cases.

**Outcome Four: 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**

All code written for this artifact was tested using iterative techniques and followed industry standard best practices in its design. The algorithms created are used in many applications in the industry and will be useful going forward in my career.

**Outcome Five: Develop a security mindset that anticipates adversarial exploits in software architecture and designs to expose potential vulnerabilities, mitigate design flaws, and ensure privacy and enhanced security of data and resources**

To meet outcome five I produced three bug free algorithms and made my code easy to maintain with object oriented design. I also focused on security by creating a password hashing algorithm to add additional security to the databases portion of my project.

Reflection:

The primary struggle I had when improving this artifact was understanding the original code and where I had left off. Because the code was unfinished I did not have a proper base to work off of and had to remember what the goals of the software were and how they were intended to be implemented. The other issue I had was with my initial plan. I had wanted to implement a sentinel linear search algorithm but found once I was working on the code that a standard linear search algorithm was a better fit.

I learned a lot about pointers and binary search trees when working on this artifact. In the initial course I really struggled with the material and never fully understood it. Working on it for the second time allowed me to fill those gaps in my knowledge and finally understand the intricacies of binary search trees and pointers.