Jael Andre

CS 499

Module 4

Professor Alim

07/28/2025

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

The artifact is a Binary Search Tree (BST) program written in C++ that I originally developed during my CS-300: Data Structures and Algorithms course in 2024. The program was designed to read municipal auction bid data from a CSV file, insert that data into a binary tree structure, and support operations such as in-order traversal and searching for specific bids. For this enhancement, I modified the BST to function as an AVL Tree, a self-balancing variation that ensures more efficient performance across all data insertion patterns.

**Justify the inclusion of the artifact in your ePortfolio. Why did you select this item? What specific components of the artifact showcase your skills and abilities in algorithms and data structure? How was the artifact improved?**

I selected this artifact because it highlights my ability to apply algorithmic thinking and data structure knowledge to improve the efficiency and reliability of a program. The original BST, while functional, did not account for balance, meaning it could perform poorly with sorted or skewed input. By converting it to an AVL Tree, I added balance factor logic, node height tracking, and four types of rotation operations to preserve optimal tree shape. This improvement showcases my understanding of recursive algorithms, complexity trade-offs, and memory management in C++. Including this artifact demonstrates my growth in applying advanced data structure techniques to optimize performance.

**Did you meet the course outcomes you planned to meet with this enhancement in Module One? Do you have any updates to your outcome-coverage plans?**

Yes, I met the course outcome I planned to address: Outcome 3, which involves designing and evaluating computing solutions using algorithmic principles while managing design trade-offs. Through this enhancement, I made a deliberate decision to increase code complexity in exchange for better runtime performance and scalability. At this point, I do not have any updates to my outcome-coverage plans because this artifact effectively supports the original objective and reflects the skills I set out to demonstrate.

**Reflect on the process of enhancing and modifying the artifact. What did you learn as you were creating it and improving it? What challenges did you face?**

Enhancing the BST into an AVL Tree taught me how to manage more advanced logic flows and make structural changes that preserve algorithm correctness. Implementing rotations required careful pointer manipulation and attention to detail to avoid disrupting the tree's integrity. One of the biggest challenges I faced was debugging issues related to recursive insertions that triggered multiple rotations—especially when balancing needed to happen at deeper levels in the tree. I also had to ensure that height values were accurately maintained to avoid miscalculating balance factors. Overall, the experience deepened my confidence with tree-based data structures and reminded me how important modular thinking and testing are when working with complex recursive logic.