Dylan W. Sanderson

CS-499-T4088 Computer Science Capstone 21EW4

Algorithms and Data Structures Narrative

The artifact I chose for the Algorithms and Data Structures enhancement was the final project from CS-260-J3682 Data Structures and Algorithms 20EW3. It was initially completed in February of 2020. This project was interesting, because we were provided with data and a parser, but had to develop various data structure methods, including sorting though linked lists, binary search trees, and hash tables. Afterwords, we completed an analysis of each of the sorting methods, demonstrating our mastery of all related topics.

I chose this artifact because it demonstrated my ability to build and understand multiple different types of sorting algorithms. Given manipulation of large data sets is vital to most companies, understanding the most efficient methods, and being able to justify said methods, of sorting and traversing the data is vital if you interact with the company’s database. In this case, the artifact was improved specifically in the BinarySearchTree.cpp file. We built methods that sorted the parsed data, and methods to find particular items, but nothing that simply printed it all in order, both proving that we sorted properly, and that we understood how to recursively navigate the data structure efficiently, so I built a ‘printAll’ method for the tree.

I did meet the objective and successfully build the method I wished to add. Given more time, I would be interested in creating a rebalancing method to ensure the tree was as efficient as possible, likely traversing the tree in a way similar to the printAll method, and outputting it into a linked list. Then, breaking up the linked list like I’m about to sort it, but rather than rebuilding it into a sorted list, put it back into the binary search tree. However, even though the scope of this enhancement was not as large as the others, it not only demonstrated what was discussed previously, it also showed my ability to return to old code and understand how it operated.

I concreted a valuable set of concepts throughout this enhancement, namely, programming best practices. It was through effective naming conventions, commenting, spacing conventions, and the like, that I was able to pick this code back up and quickly understand how it functioned, then manipulate it further. It also solidified the idea of recursion, which is an incredibly valuable programming concept that is utilized often. Without mastery of recursion, coding becomes cumbersome, ineffective, and failure-prone.