Jesse Peterson  
CS499  
Professor Bryant  
28 September 2024

The enhancement that I chose to tackle is a comparison of data structures and algorithms. Furthermore, this enhancement features the data structures, Binary Search Tree, Hash Table, Linked List, and vectors. I put all of these data structures back-to-back and compared the complexity when it comes to sorting. Now, the binary search tree is unique where it is already sorted. I compared the complexity between the other three.

The inclusion of this artifact into my portfolio showcases my ability to apply complex data structures and algorithms. This is the opportunity that I was not given during my time in DSA, I think it is the proper enhancement to prove my knowledge. Specifically, developing data structures and algorithms displays my learning in this area of computer science. The artifact has been improved by fixing the removal functions, combining all algorithms into one program, and applying sorting algorithms.

The outcomes that I believe I have met:   
   
“Employ strategies for building collaborative environments that enable diverse audiences to support organizational decision-making in the field of computer science”

“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 (data structures and algorithms)”   
   
I believe that I have met the first outcome because this work will help decision-making when it comes to the field of computer science. Next, I believe that I met the second outcome because I solved a problem using algorithmic principles appropriate to its solution.

I first started by combining the code together. Then I made a main class for running the program. After I had my program assembled, I implemented cmake for building the program with ease. After this, I implemented sorting algorithms to each of the data structures. Finally, some small touch ups were made to finalize the code. When creating it, I learned that it would be pointless to sort the binary search tree, since it is inherently sorted. To compensate for this, I decided to throw in another data structure, a vector. I faced many challenges, one of which was that my provided data was already sorted. To fix this problem, I made a program on the side that would read the file and scramble the contents. This allowed me to verify the functionality of my sorting algorithms.