**Final Analysis**

* My program works as well as my expectations. It includes the exception handling by using the while loop for the inputs that the user puts in the program, which prevents input buffer errors while using the program.
* The function to display transaction history is not available because I did not have enough time to implement it. It is complex for me to store all the transaction histories when I use text files to store accounts’ pieces of information, which can create a lot of redundant text files if more and more accounts are created. I think if I must implement this functionality, I will need to create each text file for each account and name the text file by using the account number which has a specific number. Maybe there are better implementations that I don’t know.
* I use the preorder traversal for the BST class so the time for traversing the nodes of the BST is log(n) and the time for accessing each account is O(1) - O(n). However, the time to read accounts from text files and import them into the BST object takes O(m\*logn), m is the number of accounts read from the file and n is the number of nodes in the tree before the insertion.
* After finishing the final project, I gained a better understanding of the binary search tree and its operations. The analysis and design took me a lot of time to figure out how to implement the ideas into the code and how to make it run efficiently since I had to decide what parameters to pass into each function in the program.