Assignment 0: Pointersorter.c

Joseph Klaszky

*Note: I worked on this project solo.

Overview:

Main features:

- Check for proper number of arguments.
- If there are the appropriate number of arguments, then build an empty Binary Search Tree (BST).
- Begin iteration through the input string.
    - If the program hits an alpha character:
        - update some variables to let the program know the size of the current string and where it is located in the input string.
    - If the program hits a nonalpha character:
        - check size of string:
            - If zero, then continue to next char doing noting.
            - If nonzero, then call a function called “pullString” to grab all the characters since the last nonalpha character.
- After a string has been pulled it attempts to be inserted into the BST. This part isn’t great. To sort things in the BST properly I needed to do two difference comparisons at times. The first comparison ignores case to get it to the correct branch of the BST. Then if two words are very similar it does another cmp that is case sensitive to see where the current value should be inserted.
- No duplicates are stored in the BST. If an identical value is trying to be inserted it will instead increment a variable in the struct called “num” that will be the number of times it is printed at the end.
- After it finishes with the input string it prints out the contents of the tree using a simple recursive in-order traversal.
- Finally the tree is freed using a post-order traversal.