Assignment 0: String Sorting
Coding by Taner Doganay & Liam O'Brien

The file pointersorter initially creates a node to store the text input from the user. Then with malloc, we check to see if there is enough free memory to be allocated prior to the sort. We then loop through the input string to find the words within the string using stringStart to store the starting index of a word from the input. If the user enters a non-alphabetic character then we know that it is the end of the word and we send the word to the insert method which inserts the word into the linked list. If the user does not enter any text, there are no alphabetical characters typed, there is more than one input to be analyzed, or not enough free memory is available, the program will kick back an error.

We then created a struct Node to insert words into our linked list. Our node is a char star called String and a node pointer called Next. Each node holds one of the words from the input while the nodes are then sorted alphabetically. We again check to see if we have enough free memory for the Node pointer and the previous Node pointer. Once the program validates there is enough space, it proceeds to add words to the linked list. During this time, we also run through a loop to compare words and to maintain order while inserting.