

Readme Asst0

Malik Ameer: 152006119

For Asst0 we were instructed to create a program called pointersorter. This program accepts an input string and returns each word in the string in alphabetical order (with non-alphabetic characters being delimiters). To begin with I decided to implement a linked list data structure, but instead went with creating an ArrayList that expands by one element upon every insertion. I chose this route to simplify implementation after discovering the qsort function that C provides. The ArrayList holds pointers to each word that is tokenized. My method of tokenizing the words was to simply traverse through every letter in the input string while appending the current letter to a temporary string pointer. The temp string is expanded by 2 chars in dynamic memory upon each insertion. After analyzing my program I realized that using buffers may be better since multiple calls of realloc tend to eat up time. When a non alpha character is detected from the input string scanner, the program points the next available array pointer to the temp string, and sets the temp string to NULL. This continues until the string is completely scanned. Qsort is then called, however it was necessary to create a function that wrapped the comparison argument since qsort required const variables. The sorted strings are in ASCII alphabetical order (Uppercase letters are given higher priority, followed by lowercase). This allows pointersorter to sort many languages because of the ASCII format. The strings are then printed out and freed, followed by the ArrayList struct being freed. The program handles all inputs and gives error messages for invalid inputs or inputs with no alphabetical characters. However, special escape characters will break it unless preceded by the slash character.