For this program, we were meant to sort words given in a single input string by reading it from the command line, breaking them into individual string components, sorting them alphabetically, and then printing them one per line in descending order. In order to do this, the two most important aspects to consider for this assignment were separating the words into components (disregarding the nonalphabetic characters) that could be compared to each other, and then reordering the array into a sorted form. This was done in the following way:

1. The argument string was passed into a character array, named argString, and then modified by a for loop which replaced every nonalphabetic character with a ‘ ‘ [space] character.
2. A second character array, newString, along with character array token, were created in order to tokenize the characters of argString into component strings separated by the ‘space’ delimiter, with each of these components being passed into the newString array.
3. By implementing two for loops to run through each string component in the newString array, a sorting function was created where each time the passed argument string ‘words[x]’ was compared to ‘words[y]’, if the compared function returned greater than 0, the strings would be swapped with the help of a temp character array.
4. Calling the sort function and passing newString and its length as arguments, the character array would be sorted.
5. Finally, using a for loop, the newly sorted list would be printed line by line and in alphabetical order.

To demonstrate how this would work in action, we can look at passing the command:

./stringsorter “This is5the string 2 test”

First, the argument “This is5the string 2 test” will be stored into the character array, and then be updated to remove the unnecessary characters, transforming it into “This is the string test”.

Next, this modified string is passed into the tokenizer, and the new character array holds the values newString[0] = “This”, newString[1] = “is”, newString[2] = “the”, newString[3] = “string”, newString[4] = “test”.

Then, this can be passed to the sorting function for it to be sorted in alphabetical order, and then printed as:

This
is
string
test
the