Create a project called index_program and add source files as you see fit.

This program is an exercise in the use of Strings and linked data structures to build a somewhat complex program. I've placed a text file containing the complete text of the book "War and Peace" in the handouts section of the course web page. You are to read this file and do some counting.

Count the number of occurrences of each character in the file. (The entire file)
Count the number of total words in the file where a word is classified as any string separated from other words by white space. Count the number of unique words (case does not matter for this count).
Count the number of lines in the file.

Output a sorted list of the unique words to an output file called word_count.txt where each word is listed on a line by itself along with its count.
Output a sorted list of characters along with their counts in a file called character_count.txt.

You may find it helpful to make two passes through the document to do all these counts. One pass for the character count and an additional pass for the other counts. Don't overcomplicate the character count, you can do it very easily with a small and simple array.

After your program creates the two output files you should display a prompt asking the user what word they would like to search for as well as displaying an initial text message giving the number of lines in the file, total number of words and total number of unique words ignoring case. After typing a word at the prompt your program will display the line numbers for all of the lines in the file (6 per line in formatted columns) that contain that particular word. This process should continue as long as the user chooses. Capitalization of a word should not matter for searching or counting in words, but capitalization DOES matter for your character count. Please note that your character count should include a count of every character (including invisible characters like space and new lines) occurring at least 1 time in the document. You can leave out those characters that do not appear at all (i.e. they have a count of zero).

This assignment is a fun and interesting project that has real world application.

Note: The input file is quite large, so I suggest you do all testing on a very small portion of the file instead of the entire thing. When counting characters I want to see the counts of every character in the entire file, this includes the header and footer information at the beginning and end of the file.