

CS2050 Fall 2017
Homework 5
Due: Tuesday, Oct 3, 2017

This fifth assignment will be to test and time searching methods. The objective is to create a concordance - a sorted list of words and the number of times each word occurs. Two structures will be created - the classic array and another as a linked list. Yes, you may use the Java ArrayList and LinkedList classes.

Requirements:

- Design classes appropriate to each of the lists and whatever else you need
- Ask the user to enter the input file name. Do not hard-code the file name into the program.
- The input file is lines of plain text - read in the file, put each word into an array and into the linked list.
 - Change all text to lowercase
 - Remove all non-alphabetic characters
 - If the word does not exist in the list, add it in the correct position and set the word counter to zero.
 - If the word is in the list, increment the word counter.
 - Sort the lists

For example, the first time the word "the" occurs, add this to both of the lists then set the counter the "the" in each list to zero.

Notes:

- The single quote ' is NOT to be removed. For example, the word "don't" should be treated as a single word and not as "don" and "t".
- There are many other non-alphabet and non-numeric characters that need to be removed.

Input:

The input file is the same as for HW4, in my github repo under the name:
`github.com/blanchecohen/TEST/HW4.txt`

Output:

There are three outputs required, your time log, a file and a summary to the terminal/console.

a) The output file named `HW5.txt` comes from the array (not the linked list) and prints the word and the number of occurrences on individual lines. It will look like this (two sample words):

dickens	5
the	11683

b) Output to the console is a summary and will look like this:

Filename: *name of the input file to be processed*
Number of words: *number of unique words in the file*

c) Use the definition of the time log from HW1. Plain text only!

Suggestions & Notes:

- More information on sorts - you create the compareTo method:
<https://docs.oracle.com/javase/8/docs/api/java/util/ArrayList.html>
<http://www.topjavatutorial.com/java-8/java-8-sort-arraylist-using-list-sort-method/>
- Use the Step by Step process from HW 1
- Create the class files and with stubs for each method. Compile and verify at each step the program works. Design and build incrementally. Once you have something working, go on to the next method. Don't try to do everything at once.
- The only user involvement is in `HW5.java`
- Avoid hard-coding when possible

- Remember, only **HW5.java** will contain the **main()** method.

To Turn In:

Upload to your github account the time log, the Java files and the output file for this program.

Grading:

If you don't submit a time log, you lose points.

If the output to the terminal is not exactly what's required, you lose points.

If the output file is not in the correct format, you lose points.

If the program fails when I use a different input file, you lose points.