

Due: Thursday, March 1 at 11:00 am.

NOTE: Please name the class you write for this assignment `WordCounter.java` and put it in a package called `username.hw3`

I have posted a data file, called “words.txt”, for your use in this assignment. As you will see, the file contains exactly one word per line. (We have the venerable Professor Sam Rebelsky to thank for this conveniently formatted text file.)

Write a program called `WordCounter.java` that counts the occurrences of words in the provided data file. Note that your program should work for any data file in this format and with this name (even if it has a different number of words or lines than the posted file does).

More specifically, your `WordCounter.java` should work as follows.

1. Your program should prompt the user for an integer value, and continue prompting until an integer is given. You should not assume that users will be experts at data-entry. They may enter non-integer values, and if so, your program should catch and handle the exceptions generated without crashing. *You will probably want to create a method to perform this task.*
2. The program should then read the data file and count the number of occurrences of the words therein. However, there is the following caveat. If the user enters a negative number, you should count the occurrences of the first 500 distinct words in the file. Otherwise, you should count the occurrences of the first n distinct words, where n is the number the user enters. If n (or 500) is larger than the number of distinct words in the file, you should of course stop when you have considered every word in the file.
3. Lastly, the program should print a list of the words for which counts were generated, along with the associated word counts. The list should be printed in the order in which the words first occurred in the data file.

Hint: You may find it helpful to keep two “parallel” arrays to solve this problem: one for words, and one for counts. The two arrays should be made to work together, such that elements with corresponding index numbers are associated with each other.

An example run of the program might look something like this.

```
Enter max number of (distinct) words to process: q
Illegal value entered (non-integer).
Enter max number of (distinct) words to process: 10
1 every
1 sunday
10 as
4 soon
20 they
3 were
1 free
36 the
3 little
1 soldiers
```

The last few lines from another example run, in which the user entered `-1` might look something like this:

```
1 first
1 clump
1 cut
```

1 stick
1 whittling
1 slowly
1 thinking
1 folks

Submission instructions:

Please submit `WordCounter.java` by sending it as an email attachment to submit152@cs.grinnell.edu. Your submission should arrive by the due date and time.