Homework 3

Due: May 25, 2023   **Points:** 100

1. (*40 points*) Define a "word" as a maximal length sequence of alphanumerics. For example, in the line

   ```
   Hello, there! How are you? I am fine; how are you? My number is DE4-3656.
   ```

   the words are "hello", "there", "How", "are", "you", "I" "am", "fine", "how", "are", "you", "My", "number", "is", "DE4", and "3646".

   Write a program that reads lines from a file (you may assume it's a text file) or files named on the command line, breaks them into words, stores the words in order (that is, ASCII order), and counts the number of times each word appears. Do ***not*** make any assumptions about the number of lines or words; but you may assume no word is over 256 characters long. When this is done, print the words and counts, as follows. This example is from the line above:

   ```
   3646 (1)
   DE4 (1)
   Hello (1)
   How (1)
   I (1)
   My (1)
   am (1)
   are (2)
   fine (1)
   how (1)
   is (1)
   number (1)
   there (1)
   you (2)
   ```

   Call your file *wordsort1.c*.
   Hint: This is easiest done using a linked list, with each element a structure containing a pointer to the word and count.

2. (*40 points*) Now modify the *wordsort1.c* so that it prints the words in dictionary order rather than ASCII order. Dictionary order, sometimes called lexicographic order, intermixes upper-case and lower-case letters, so the upper-case letter precedes the corresponding lower-case letter. Digits precede all letters.

   Here is the same list as in the example in problem 1, but in lexicographic order:

   ```
   3646 (1)
   am (1)
   are (2)
   DE4 (1)
   fine (1)
   Hello (1)
   How (1)
   how (1)
   I (1)
   is (1)
   My (1)
   number (1)
   there (1)
   you (2)
   ```

   Call your file *wordsort2.c*.

3. (*20 points*) The program *calc.c* has two bugs.
   (a) At the "operand" prompt, type any non-integer. The program will go into an infinite loop.

(b) Carry out any of the operations of the calculator. The operation will be performed, but you will then see a spurious message "invalid operator".

Find and fix the bugs.

Call your corrected file "calcans.c".