EE231002 Introduction to Programming

Lab 24. Word List

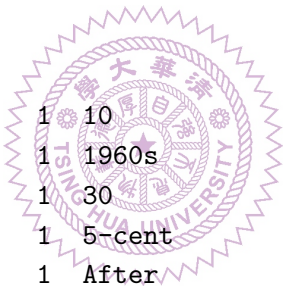
No due date.

In this lab, you will write a program to list all the words used in a text file and print out the number of times each word has been used.

It is suggested that you create a linked list to store all the words using the following structure.

```
typedef struct sWORD {
    char *spelling;     // the word spelling
    int count;          // number of times the word has been used
    struct sWORD *next;
} WORD;
```

It is also suggested to order the linked list lexicographically for easier searching and sorting. The example output for the text file `Jobs.txt` is shown below.



```
$ ./a.out < Jobs.txt
-                9  000              1  10               4  12               1
17               2  18               1  1960s            1  2                1
20               1  2005             1  30               3  33               1
35               1  4                1  5-cent           1  6                1
7                2  About            1  After            1  Again            1
And             18  Apple            8  Apple's          1  As               1
Because          2  Beneath          1  Board            1  Bob              1
Brand            1  But              5  Catalog          2  Coke             1
College          2  David            1  Death            1  Directors        1
  ......
with            19  woman            1  wonderful        2  words            1
work             5  worked           1  working-class    1  world            2
world's          1  would            9  year             3  years            9
yet              2  you             30  you'd            1  you'll           2
young            2  your            14  yourself         1
```

It is shown that the word `"And"` has be used 18 times and `"Apple"` eight times, etc. Capitalized words are different from lower case words. And, the print out is in lexicographical order (which is the result of using `strcmp` function for comparison.)

Notes.

1. A word starts by a alphanumerical character (`'0'..'9'`, `'A'..'Z'`, `'a'..'z'`) and also ends by a alphanumerical character. Special characters such as `'\''` (apostrophe), `'_'` (underscore) and `'-'` (hyphen) are acceptable within a word. Other characters, such as `' '` (space), `'\t'` (tab), `'\n'` (new-line) and all the punctuation characters cannot be part of a word.

2. The text file, `Jobs.txt` can be used to test your program. Of course, you should also try out with other text files as well.

3. The output example shown above list the words in lexicographical order, and capitalized words are treated differently from the lower case words. More programming practice can be done as the following.

   3.1. Treat capitalized words in the same way as lower case word. For example, `All` and `all` are treated as the same word and they are printed out only once with sum of their usages as the count.

   3.2. Print out the words by the number of usage. The most used word print out first, followed by the second most used one. If a number of words have the same number of count, then they are printed out using lexicographical order.

4. You can submit your codes by

   ```
   $ ~ee2310/bin/submit lab24 lab24a.c
   $ ~ee2310/bin/submit lab24 lab24b.c
   $ ~ee2310/bin/submit lab24 lab24c.c
   ```

   The three files are for different output listings.