資料結構Project1

Dr. Mao-Mao and X Company‘s vault

1. Description

Dr. Mao-Mao is a very hard-working professor; however, he becomes a master thief who robs the rich and assists the poor at night.

This time, Dr. Mao-Mao has declared war to the evil X Company, which sells fake products to fool customers, and earns a lot of black gold. For justice, he MUST rob X company’s money to save everyone from suffering. He hacked into X Company’s database and found every X Company’s vault has an encrypted document. At first, he had no idea what did they mean. Three days later, he finally found a tricky method to crack the documents!

Dr. Mao-Mao found that the first word in each document is the keyword, and that he could use the keyword to search other keywords in the document. If he finds any word that contains the keyword, he could record the number of this word from the beginning and keep searching until the end of the document. Finally, after putting all the record number together in a row, it is the password to open the vault!

Since there are too many documents to crack, Dr. Mao-Mao needs your help to find every secret password in each document. Can you help him?

(it’s Okay if you don’t read this section. It’s too hard for us either.)

2. Input / Output

I. Input (From file)

An article in extension of txt, which means the nonsense document. Each word in article is separated by space, punctuation marks or new line. The document’s characters will not greater than 1,000,000.

II. Execution Process(Standard input)

Input_file_name.txt Output_file_name.txt

There will be only one line each test. The line indicates the input filename and output filename of the article, separated by a space. Your code should be able to use them for accessing content of article.

see Example 1~3 below in detail.

III. Output(File output)

Output two lines each test. In the first line, show the running time in the unit of

microsecond (from the time after keying the input to the time your program ends). And in the second line, show the password of the vault. Notice that the number of the word which contains the magic code is begins from 1. Also, if a word contains the magic code more than once, show the number of the word which times it contains.

See Example 1~3 below in detail.

Example 1.

Input file:
at train station the cat at ate a bat, while another cat at the corner attacked by a bat.
Execution:
test1.txt test1out.txt
Output file:
1ms 13567912131619

Example 2

Input file:
can you can a can as a canner can can a can? cancancan
Execution:
test2.txt test2out.txt
Output:
1ms 135891012131313

Example 3

Input file:
aa aaa aaasaaaa
Execution:
Test3.txt test3out.txt
Output:
0ms 12233333

3. Requirements

Program

I. You need to turn in the code.

II. Name your code file "hw1_StudenyID.c/cpp." (Ex. hw1_0316000.c/cpp)

III. Using File output to print out your results to output file and record them in your report.

IV. Your program must be readable (Ex. Comments, variable names, function names)

Report (Name the file"hw1_StudentID.pdf", **Ex: hw1_0316000.pdf**)

I. Describe your implementation. (Ex: algorithm, program executing process)

II. Sample testing Results: the execution result for sample testing (show it by taking screenshots)

IV. No more than 2 pages.

4. Grading policy

I. Sample testing (60%)

Your code can run without crashing and giving correct results to the sample files in your report.

II. Performance (20%)

Judge by your executive time. The better of your performance, the higher is your score.

III. Implementation describing (20%)

IV. Bonus (10%)

using failure function to match string.

5. Submit (e3 will be closed on time)

Compress all your files (including your code(.c/.cpp) and report(.pdf)) Name your compressed file "hw1_studentID.rar" or "hw1_studentID.zip". Upload your compressed file to e3.

Deadline: 2015.10.26, 23:59
No late upload.