CSCE 3600: Systems Programming

Minor Assignment 2 – File I/O, Memory Allocation, Parsing

Due: Tuesday, September 18, 2018 at 11:59 PM

PROGRAM DESCRIPTION:

In this assignment, you will write a complete C program that will parse an input file and search for occurrences of a word pattern. The program will output the total number of occurrences and all the lines with the word pattern in it.

- **Command-Line Arguments**

  Read in the command-line arguments and verify that there are three parameters: (1) the program name, (2) an input file that is the name of the file you will be working with, and (3) the word pattern that you are searching for. If the number of parameters is incorrect, print out a usage message and terminate the program.

- **File I/O**

  Open the file that was given as a command-line argument to your program. Then, read through the file and print the following information: (1) the total number of lines in the file, (2) the number of characters in the longest line of the file, and (3) the longest line of the file.

- **Pattern Matching (i.e., grep)**

  Create a dynamic data structure that holds all of the information for each occurrence of a word pattern. For example, it should hold the line number and the contents of the line it occurs on. Print all of the information (i.e., the line number and the contents of the line it occurs on), including the total number occurrences of the word pattern.

Some assumptions and general rules:

- Your program should be case sensitive (i.e., the word pattern “and” is different than the word pattern “AND”).
- You may assume that no lines in the file will be longer than 80 characters (81 with the null character). You may assume that the lengths of the filename and word pattern given as command-line arguments are not longer than 15 characters (16 with the null character).
- You may not make any assumptions about how many occurrences of the word pattern are in the file, or each line for that matter. Therefore, you may want to use a dynamic array or linked list and allocate memory for each occurrence as you go. You must deallocate the memory when no longer needed.
- Add error checking where needed (e.g., after opening file, allocating memory, etc.) and take the appropriate action.
- If you have any questions about the functionality, please contact your instructor.

REQUIREMENTS:

- Your code should be well documented in terms of comments. For example, good comments in general consist of a header (with your name, course section, date, and brief description), comments for each variable, and commented blocks of code.
- Your program should be named “**`minor2.c`**”, without the quotes.
- Your program will be graded based largely on whether it works correctly on the CSE machines (e.g., `cse01`, `cse02`, …, `cse06`), so you should make sure that your program compiles and runs on a CSE machine.
- This is an individual programming assignment that must be the sole work of the individual student. Any instance of academic dishonesty will result in a grade of “F” for the course, along with a report filed into the Academic Integrity Database.

SAMPLE OUTPUT (user input shown in **bold green**)**:**

```
$ gcc minor2.c
$ more input1.txt
This life, which had been the
tomb of his virtue and of his
honour, is but a walking
shadow; a poor player, that
struts and frets his hour upon
the stage, and then is heard
no more: it is a tale told by an
idiot, full of sound and fury,
signifying nothing.
     -- William Shakespeare
$ ./a.out input1.txt and

Total Number Occurrences of "and" in File: 4
-----------------------------------------------------
Line 2: tomb of his virtue and of his
Line 5: struts and frets his hour upon
Line 6: the stage, and then is heard
Line 8: idiot, full of sound and fury,
-----------------------------------------------------
Total Lines in File      = 10
Characters Longest Line  = 33

Longest line             = no more: it is a tale told by an

$ ./a.out input1.txt his

Total Number Occurrences of "his" in File: 4
-----------------------------------------------------
Line 1: This life, which had been the
Line 2: tomb of his virtue and of his
Line 5: struts and frets his hour upon
-----------------------------------------------------
Total Lines in File      = 10
```

```
Characters Longest Line   = 33
Longest line              = no more: it is a tale told by an
```

SUBMISSION:

- You will electronically submit your program to the **Minor Assignment 2** dropbox in Canvas by the due date and time.