Lab 6

Deadline 11:59PM Nov 4

Descriptions

This lab is a good practice to use Github for collaborative purposes. You are required to write a program that reads the content of a file. The number of lines in the file is not known (it could be thousands of lines). The format of each line is:

FirstName LastName GPA Status TOEFL

Example (student.txt)
Mary Jackson 4.0 I 60
Jack He 2.45 D
….
Mike Johnson 3.125 D
Jane Zhang 3.8 I 120

Your program should read all the lines and store each line as a struct named DomesticStudent and InternationalStudent.

Requirements

1. Your program should then write a list of students whose GPA is greater than 3.9 to a file.
2. An international student has a TOEFL score ranging from 0-120. If the international student has a TOEFL score less than 70, then it should not be store to a file.
3. DomesticStudent struct does not have the TOEFL field.
4. I stands for an international student and D stands for a domestic student in status column.
5. The order of students must be the same order as in the original file.
6. If the format of a student does not conform the specified format, your program must print the appropriate message and exit. This means no output file should be printed.
7. The first argument is the intput file namd and the second argument is the output file name. The run command will be ./a.out <input file> <output file> <option>
8. The option field is the command line is an integer (1, 2 or 3). Here are descriptions of options.
    a. 1 only saves the filtered output of domestic students (no international students in the output file)
    b. 2 only saves the filtered output of international students (no domestic students in the output file)
    c. 3 only saves the output of all students
    d. All options must output only those students that meet the criteria described in above requirements.
9. You must handle corner cases with reasonable output messages (e.g. "Error: XX")
10. Assume that everyone has a first name and last name.

11. Make sure to run your code in debug run. Your code must exit with exit code 0. A code failed to run due to not checking in debug mode will result in penalties.

Grading

This lab will be marked out of 9. For full marks this week, you must:

- (1 point) Correctly use git/GitHub and the repository following the handout
- (4 points) Generate a correct solution to the problem(s) in this lab
- (2 point) Handling corner cases and gracefully exit
- (1 point) Reasonable division of functions into a separate c file
- (1 point) Reasonable comments that explain functions and variables

Submissions

1. Github link is posted on Learning Hub
2. lab6.c (lower case)
3. file_handler.c
4. file_handler.h
5. AXXXX.txt (empty file, but with you're A number as file name)
6. Your main function must be in lab6.c.
7. If you have any other auxiliary files (.c and .h), please include them in thie github as well.
8. Do not use subdirectories. All files must reside in the top source directory.