CCPROG2 MP PART 3 [20% of the MP] *サルバドル・フロランテ*
AY2023 Term 1

GENERAL INSTRUCTIONS

1. Honesty Policy and Honor Code apply. Please do not cheat. **The consequence of cheating is a grade of 0.0 (fail) for the course.**
2. Make sure that you have read and understood the problem requirements.
3. Comply with the specifications and restrictions stated in MP Specs document, and inside the comments written in the skeleton codes. Non-compliance will result in deductions.
4. **You are NOT allowed to use pre-defined library functions that we did not discuss in class (unless specified otherwise).**
5. **Do NOT use float data type. Always use double data type for floating point numbers in all MP challenges.**
6. Subject your solution to exhaustive testing to ensure that the solution is logically correct. The following scoring and deductions system will be applied.
    - A perfect score will be awarded to a compliant and logically correct solution.
    - Deductions will be applied based on the severity of the logical error. In the worst case, scenario, a score of 0 will be given (meaning the solution is logically incorrect).
    - Each unique compiler warning will result in a deduction of one point. Do not forget to use -Wall compiler option.
    - A syntax error will result in a score of 0 for the associated function (as answer to the question).
7. Submit the required deliverables before the specified Canvas deadline.
8. Question? Please post your question in our Canvas MP Discussion Thread. Note: I will not answer an MP related question sent via email unless it is personal in nature (for example, you and your partner would like to part ways because of disagreements in working habits).

This is the last part of the MP. The activity numbers are continued from the previous part.

For MP Part 3: choose which set of questions to answer, i.e., choose either your own five questions, or those from another group that you answered in MP Part 2.[1] You must re-implement the functions by applying file processing. Specifically, you have to use **fopen()**, **fscanf()**, **fprintf()** and **fclose()** functions.

ACTIVITY #8: Re-implement the functions using text file processing.

1. You'll have to retain the header file **LASTNAME1_LASTNAME2.h**, and re-use all its contents, i.e., **#define**, **struct** data type declaration(s) and **typedef** alias declaration(s).
2. Encode in the accompanying skeleton C source file **LASTNAME1_LASTNAME2.c** the functions that will produce the answers to the 5 questions that you chose. The following are hard requirements that should be present in your C source file submission:
    - Data must be read from the **SoGA_DATASET.txt** file using **fscanf()**. All the contents of this file should be stored into a 1D array of **struct** just like in MP Part 2.
    - The main() function should call the appropriate C function, and then call **fprintf()** to write the answer to each question onto the **OUTPUT_LASTNAME1_LASTNAME2.txt** file. Numeric answers with double data type must be written with 6 digits after the decimal point.
    - There should NOT be any **printf()** andr **scanf()** statement in the entire source code, not even in main(), and in the function that reads the SoGA data text file.
    - You must use a C **double** data type (NOT float) for all floating point values/variables/parameters/functions.
    - Make sure that the answers are NOT HARDCODED inside the function definitions.
3. Submit three files, specifically (i) C source code file, (ii) header file, and (iii) the output text file of your executable file via the Canvas submission page for this purpose.

ACTIVITY #9: Demo your MP.

We will schedule a date/time for you to demo your project to check if it is working or not. You should know how to compile and run your program in the command line. You and your partner must both be present. The demo will be done in person *(but may be set to online mode in case of time/venue/resource related issue)*.

--- End of MP Part 3 ---

---

[1] My recommendation is for you to re-implement the functions that answer your own 5 questions. You may start doing MP Part 3 as soon as you're done with Part 1. To do this though, you should not wait for me to cover the topics on struct data type and file processing. You'll have to learn these in advance and on your own to accomplish Part 3 well ahead of time.