ELET 4309 (C++)
100 Points | **Homework No. 7**
Due 04/29/2019 | **Dr. F. Attarzadeh**
04/18/2019

Final Exam
ELET 4309, Spring 2019

8AM-10AM, Tuesday, May 7, 2019

Room 116A-T

It is very important that you read the notes at the end of each homework assignment for this and all other assignments. Make sure that you read related lessons in the notes, examine sample programs, study the exercises, and study the chapters in the book. These are all prerequisites for better understanding of the files and file classes.

Develop a functional flowchart and then write a menu-driven program as explained below. You will be using files, file-related concepts and exceptions. The requirements are explained below.

There are many places where a try and appropriate handlers (catch blocks) can be inserted.
The menu for this assignment is shown below.

Help **N**ame **C**ompany **A**ll **Q**uit

Once the program is executed, the above menu will be displayed at the top of the screen, and the user is prompted for a menu item selection.

H or **h** (for **H**elp) option when selected, will open a help file (this is a text file), which briefly explains how the program operates. The help screen(s) will remain on the screen until the user strikes any key. The program will then return to the main menu. For the name of this file, use the initials of your name followed by help.txt (e.g., My file will be named **FAh.txt**). The file will be closed at the end of this option. You may create a read-only file using any text editor.

N or **n** (for **N**ame) option, once selected, will first prompt the user for the name of the **Name** file and creates the file. The program will prompt the user again and asks for your full name. Once data is inputted, the program writes the data to the file and displays **Data written to the file successfully**, followed by the message *Strike any key to continue….* Striking any key will cause the screen to clear and then followed by the menu display. For the name of this file, use the initials of your name followed by **n.txt** (e.g., My file will be named **FAn.txt**). The file will be closed at the end of this option.

C or **c** (for **C**ompany) option, once selected, will first prompt the user for the name of the **Company** file and creates the file. The program will prompt the user again and asks for the company name. Once data is inputted, the program writes the data to the file and displays **Data written to the file successfully**, followed by the message *Strike any key to continue….* Striking any key will cause the screen to clear and then followed by the menu display. For the name of this file, use the initials of your name followed by the **c.txt** (e.g., My file will be named **FAc.txt**). The file will be closed at the end of this option.

A or **a** (for **All**) option, once selected, will first prompt the user for the name of the **All** file and creates the file. This file then reads the contents of the other three files, in the sequence of **Help**, **Name**, and **Company** and writes them to this new file (one per line), displays **Data written to the file successfully**, closes all the files and then opens this new file, reads its content and displays them on the screen, followed by the message *Strike any key to continue….* Striking any key will cause the screen to clear and then followed by the menu display. For the name of this file, use the initials of your name followed by the **a.txt** (e.g., My file will be named **FAa.txt**). The file will be closed at the end of this option. The other three files (i.e., name, and company) must have been created for this option to function properly. Your program needs to check for the existence of other three files and if any file is not there, write an error message and allow the user to create the missing file(s) without terminating the program.

Q or **q** (for **Q**uit) option, once selected, will clear the screen and return the control to the IDE.

Notes:(please read very carefully)

1. Make sure your media is **VIRUS FREE**!(grade of 0 will be given for infected media). Use Technology lab PCs for the test.
2. Comment your program.
3. Use meaningful prompts
4. Provide a brief description of the problem being solved
5. Be sure to include a header file at the beginning of your program as shown in the course syllabus
6. **NO global declarations of VARIABLES and *const* members allowed** in any program that you develop in this course.
7. Use file concepts and related topics. You may use other concepts covered so far.
8. Full function prototyping is required for both the member functions and non-member-functions. All functions must have their purposes fully explained.
8A. No member function should be defined within a class (i.e., no body of a function should be seen inside any class)
8B. Full member -function prototyping is required. Member functions must have their purposes fully explained.
8C. Member functions may call other member functions, as needed, to facilitate modular programming.
9. Make sure to use constructor(s) and virtual destructors for all classes used. In fact you can create a file class and start from there.
10. Parameter passing to functions will be ***by reference or address,*** and return value from functions will be ***as specified in the member function prototypes.***
11. Use data types as specified above (mostly text in this case). Your files should be readable in any text editor.
12. At the due date submit **H7.CPP and all related files for your program. Other files will be submitted as needed, and the project file,** and **H7.EXE** files in its root directory, program flowchart, source file, and *sample runs* of your program. Please review how to submit your work to the Blackboard shown in the class. You also need to submit the Visio file for this assignment.
13. Use **Microsoft Visio 2013** to develop your flowchart.
14. Use **Microsoft Visual C++ Enterprise 2015** compiler using default compiler settings.
15. Adherence to the ***ANSI C++*** is required.
16. Homework is due at 9PM of the due date. L***ate homework will not be accepted***.
17. Use formatting specifications as discussed in class.
18. *arrays* may be used for this assignment.
19. No ***goto*** statements allowed in any program that you develop in this course.
20. ***Do not*** use **<stdio.h>, <cstdio>** and **<conio.h>** in this assignment and any future assignments.
21. ***Do not*** use any **#define** in your program, other than those needed for the class header file.
22.
23. No collaboration on this assignment and all other assignments allowed. If you violate this policy, your grade for the course will be **F**.
24. ***Non-compliance with these notes will cost you points***.