VG101 Final Project: Student Management System

<u>Assigned: 15 April 2020</u>

<u>Due: 29 April 2020</u>

By using MS Visual Studio, you should implement the following program with C or C++.

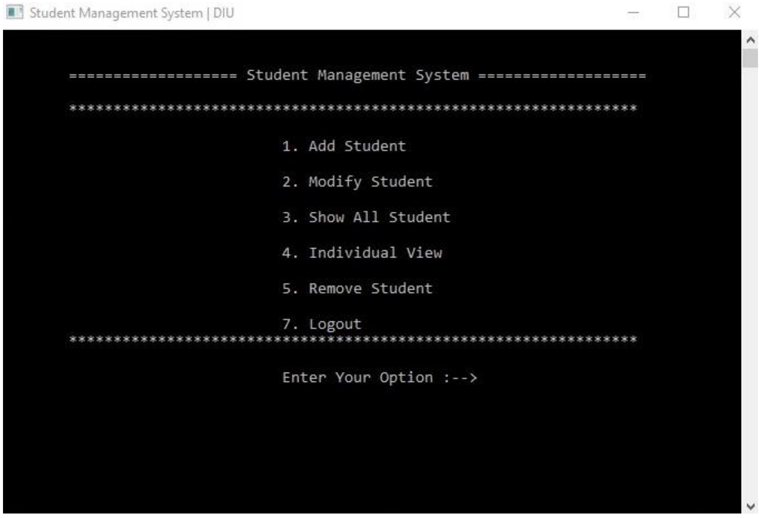
Introduction

Design a Student Management System (SMS) to store information of students. The basic requirements for this SMS project cover functionalities of adding records, modifying records, showing all records, finding individual record (based on name, number, etc), remove record(s), etc. (as shown in Fig 1 as a sample main menu). In each option, you may add one or more sub-menus. For instance, you can 1) delete all records, or 2) delete individual record in the option 5, You should also make a few more fool-proof functions. For example, the ID number should only allow 0-9.

For storing information, we use binary or text file(s) as a database. We can assume that each student record has the following fields: ID number, name, gender, age, grade of math, computer and English. Since it is based on C/C++, you are free to use OOP or Procedural Programming or the mixture of both.

Of course, you are free to add more functions. For instance, we can add function 6, count the number of marks, in which we have learnt to do simple printing about the number of students in certain ranges of marks. You can also implement binary search or binary tree search to find the records.

Fig 1. A sample of SMS's main menu



```
=================== Student Management System ===================
******************************************************************

                         1. Add Student

                         2. Modify Student

                         3. Show All Student

                         4. Individual View

                         5. Remove Student

                         7. Logout
******************************************************************

                         Enter Your Option :-->
```

Should you want to get a distinction, you should add at least two more functions to the system that will make it more practical and useful for a university teacher/staff. You may seek the opinion of your subject teacher/TA before you do so. In principle, it must be some new features that require additional efforts but not a repetition of work. A few samples of optional functionalities for your references:

- Provides the searching options based on various search factors. Such as Profiles, Student Name, Courses, Exams Marks;
- Different Login options for staffs or students;
- Report Generation;
- Provide filter reports on Student, Courses, Exams;
- Etc.

Report

Your report should include:

1. **Introduction**: A detailed description of the objectives and requirements of the project, and a brief description of the methodology.
2. **Methodology**: It contains
   - The data structure of the program developed, including
     - The specifications of the essential structures/classes defined for the data, and the public/private member functions/variables inside - explain as far as possible why you makes such choices.
     - The flow of execution of each function. (It is good to include a flow chart to illustrate it.)
   - What technical problems you encounter, and how you tackle the problems
   - Testing of your program, which shows how you validate your program, i.e. confirm that the solution is correct.
3. **Results**

   Include the results of executing your program captured from the screen.
4. **Conclusion and future development**

- Summarize the experience gained in the project
- Indicate how your program can be extended and improved if more time is allowed.

In addition, a User Manual of your program should be attached to the report as an Appendix to illustrate the usage of your program.

The report should be in PDF format. It is NOT required to include the complete source code in the report. Rather, you should copy the folder(s) containing all your project file(s), which also stores the report.

General Description

1. Unless you get prior approval from your subject teacher/TAs, you must observe the following:

- Do NOT use any technique or C/C++ constructs not taught in the subject
- Any library not mentioned in the subject must NOT be used.

2. After finishing the project, you should upload one ZIP/RAR file that contains the following:

- The soft copy of the report (including the Appendix of User Manual)
- All folder(s) and file(s) of your project
- A readme file (readme.txt) that tells us how to build the project and, if any, other important (IT) requirements for running the program. Note this is not a replacement of the User Manual which offers more comprehensive description about your system.

3. The documentation for your final project is a very important part. The ability of writing good comments will also be an important factor to the final assessment of your final project.

4. It is compulsory to use a word processing tool to write your report. The font size must not be bigger than 12 or smaller than 10. Use 1.5 lines spacing on both sides of a page. Including all diagrams and tables, if any, the length of the report should not be longer than 15 pages.