CS2910 - Assignment 1 (20 points)

Application development based on the traditional file processing approach

Week 2

Description of tasks involved in the assignment

Write an application that works with the following data:

- student contact information (id, last name, first name, phone, email);
- information about courses (name, code, semester);
- information about all grades in all courses (taken by students, different students can have different numbers of grades, depending on the number of courses taken by a specific student).

Data is stored in 3 different CSV(comma-separated values) files.

The application has the following list of features (1 point per feature):

1) output original list of all students;
2) output list of all students sorted by name (in alphabetic order and vice versa);
3) output original list of all courses;
4) output list of all courses for the specific semester (specified by the user);
5) output list of all courses for the specific semester sorted by course name (in alphabetic order and vice versa);
6) add new student, add new course, add a grade for the course (user enters student's last name, course code, and grade);
7) update student info (user enters student id);
8) search for course by name (successful result: full course info);
9) search for course by code (successful result: full course info);
10) search for student info by last name (successful result: full student info);
11) search for student info by last phone number (successful result: full student info);
12) output list of all courses taken by student (user enters student last name) and grades
13) output list of all courses taken by student (user enters student last name) and grades
14) calculate average grade for a specific student (user enters student last name)
15) calculate the average grade for a specific student for a specific term (user enters student's last name and term)
16) calculate average grade for specific courses (user enters course name).

Note: *after any action involving data modification, the contents of the corresponding file(s) must be updated.*

Selecting the most suitable data structures – 2 points;
UI - 1 point.
File processing – 1 point.

Outcomes (should be submitted in the myClass in ZIP format):

- all sources files (format depends on the programming language selected by the student);
- a shared (with the instructor) link to the project on the online IDE (if online IDE was used for the development) – optionally;
- screenshots (in jpg format) of test cases for all required features of your program.

This is an individual assignment. Even partially copied code will be subject to regulations against academic integrity. Do NOT share your solution with anybody.

Late submissions are penalized 10% / day (1 point / day), up to 3 days.

Submissions will not be accepted 1 week after the due date.