Student Assessment System

Programming Principles-1 Final Project (Spring 2019/2020)

Lecturer: Dr. Hamed Abdelhaq

Objectives:

In this project, you need to build a "Student Assessment System" that assists lecturers to manage the grades of students in a course and collects some needed statistics on these grades. Using this system, the lecturer will be able to:

1. Store the student IDs, full name, and their grades in a file "course.csv"
2. Load the content of this file
3. Update the names or/and the grades.
4. Collect some statistics on the grades
    a. The total grade of a student
    b. The average grade in a certain exam for all students
    c. The average of the total grade for all students
5. Display student records according to the total grade in descending order.

File Structure:

The file "course.csv" has a number of 20 records, where each record has the following comma-separated fields:

St_no	St_full_name	St_grade_1	St_grade_2	St_grade_3

Some examples of these records are:

```
11923113, Ahmad Akhalil,20,25,30
11834221, Hanadi Khalid,19,22,35
.
.
.
```

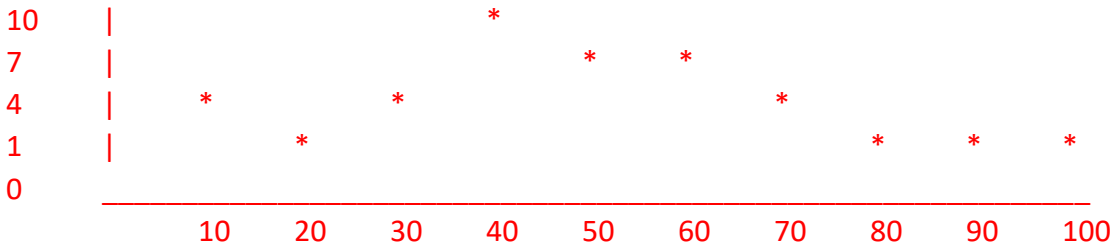
Note that

- each record should be placed in a separate line.
- The first two grades are out of 30, and the final grade is out of 40. Thus, the total grade for each student is out of 100.

Functionalities:

When you run the program, it directly loads the content of the file and stores it in appropriate array of "struct" records. Then, the program prompts user to enter one of the following items:

1. Print the content of the file on screen
   [The application displays the content of the file on screen]
2. Print the total grade of a student
   [The application prompts user to enter a student ID, then it prints the sum of his/her grade in the 3 exams]
3. Print the average grade in a certain exam
   [The application prompts user to enter exam number, then it prints the average grade of that exam for all students]
4. Print the average of total grade for all students
   [The application computes the total grade for all students, then it computes and prints the average of the stundents' grades]
5. Update student information
   [The application prompts user to enter a user ID, it displays his/her information (id, grade_1, grade_2, and grade_3), and asks for new information for each field. When the new information is entered, the array is updated accordingly]
6. Store the updated content of the array into the file
   [Store the content (that might be updated) of the array into the same file "course.csv". Note that the original content of the file will be deleted and replaced with the new content]
7. Reload the content of the file
   [populate the content of the array again with information read for the file "course.csv". Note that the content of the array will be replaced with the new content from the file]
8. display records according to the total grade in descending order.
   [You have to display the content of the array starting from the student with the highest total grade to the one with the lowest grade]
9. draw a histogram to show the distribution of total grades ➔ (bonus of 3 marks)
   [A histogram will be drawn on screen to depict the distribution of total grades. The x-axis corresponds to grade bins and the y-axis shows the frequency of grades in each bin. You can use a bandwidth of 10 for the histogram]



For example, the above histogram shows that the number of students who get a grade in the range (60-70] is 4.

Rubric:

item	Description	ratio
1	Each one of the above required points (9%)	72%
2	The use of “struct” type	8%
3	Well-organized code (using functions, no code repetition)	10%
4	Running without syntax errors	10%

A sample output:

Student Assessment System

The application provides the following services:

1. print the content of the file on screen
2. print the total grade of a student in an exam
3. Print the average grade in a certain exam
4. Print the average of total grade for all students
5. Update student information
6. Store the updated content of the array into the file
7. Reload the content of the file
8. display records according to the total grade in descending order
9. draw a histogram for total grades
10. Exit

Your choice is: (assume user enters 5 and then presses enter)
Please enter student ID: (assume user enters 11923113 and then presses enter)
Student Name:
Ahmad Akhalil ==> new name: (assume user enters “Ahmad Khalil” and then presses enter)
Grade 1:
20 == > new grade: (assume user enters “21” and then presses enter)
Grade 2:
25 == > new grade: (assume user enters “24” and then presses enter)
Grade 3:
30 == > new grade: (assume user enters “33” and then presses enter)

The application provides the following services:

1. print the content of the file on screen
2. print the total grade of a student in an exam
3. Print the average grade in a certain exam
4. Print the average of total grade for all students

5. Update student information
6. Store the updated content of the array into the file
7. Reload the content of the file
8. display records according to the total grade in descending order
9. Exit

Your choice is:

Rules:

- Each student should work alone on this project.
- Any detected cheating will result in “zero” grade for both parties, the one who gives his/her solution and the other who takes the code.