**CIS 1111 Programming Topic 09 – Averaging Quiz Grades**

**(20 Points)**

**Description:**

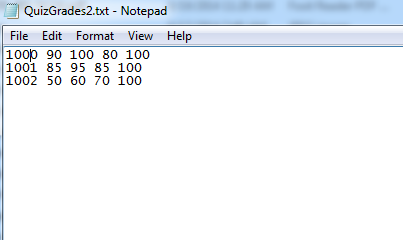
In this assignment, you will write a program in C++ that uses files and nested loops to create a file from the quiz grades entered by the user, then read the grades from the file and calculates the average quiz grade for a class. Each student takes 4 quizzes. Use a nested loop to write each student’s quizzes to a file. Then read the data from the file in order to display the student’s average score and the class average. Also write the program in Raptor. Input the data into the Raptor program during runtime. Do not use files for the Raptor program.

Enter 5 items for each student. Column 1 is the student ID and Column 2-5 are the quiz grades.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Student id | Quiz 1 | Quiz 2 | Quiz 3 | Quiz 4 |
| 1000 | 90 | 100 | 80 | 100 |
| 1001 | 85 | 95 | 85 | 100 |
| 1002 | 50 | 60 | 70 | 100 |

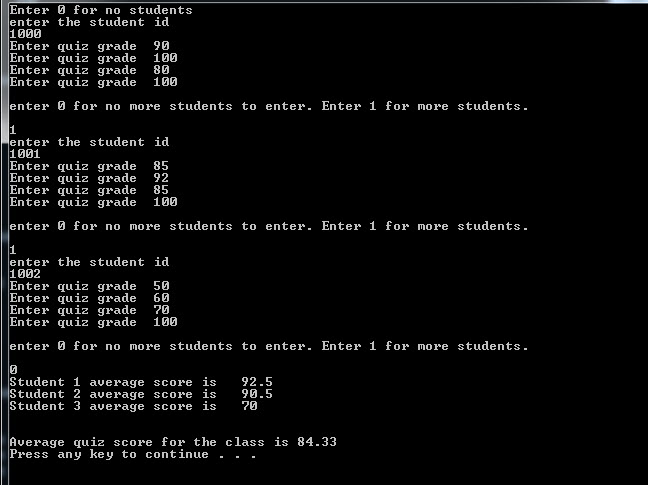
**Requirements:**

1. Use a nested loop structure to input the data and Write the data to a text file.
   1. The outer loop will be a while loop; the inner loop will be a for loop (4 quizzes).
   2. Validate whether the quiz scores are in range (0-100).
   3. Since you do not know how many students will be entered, add a way to quit the loop.
   4. Add spaces in between each item added to the text file. Add a new line after each student. The text file that your program creates will look like the following



1. Use a nested loop structure to read the data from the text file and calculate the student’s average grade.
   1. The outer look will be a while loop; the inner loop will be a for loop (4 quizzes)
   2. To calculate each student’s average score, use a total variable initialized to 0 before the for loop, then calculate the student’s average after the loop.
   3. To calculate the class average, initialize a classTotal variable to 0 before the while loop, add each student’s total into the classTotal following the for loop, then calculate the classAverage after the while loop.
   4. Only display 2 decimals for the averages.
2. BEWARE of integer division.
3. Use cout to output the values of the variables to the console. Your cout statement MUST use the variables to display the values
4. Output must be labelled and easy to read as shown in the sample output below.
5. Program must be documented with the following:
   1. // Name
   2. // Date
   3. // Program Name
   4. // Description
6. Write the program in Raptor. Input the data during run time.

**Sample Output:**



**Submit:**

Zipped folder named LastNameFirstNameCIS1111NameOfAssignment which contains:

1. Your .cpp file
2. Screen shots of your code and output
3. The Raptor program
4. Screen shots of the Raptor program

**Grading Guidelines for This Assignment**

|  |  |  |
| --- | --- | --- |
|  | **Range – Low End**  **(Did not do or did very little effort)** | **Range – High End**  **(Used correctly and spent time/effort on programming)** |
| Names of variables are meaningful and the program comments self-document the program | 0 | 2 |
| Met all stated requirements | 0 | 10 |
| Output is correct given the input, and the output is correctly formatted | 0 | 4 |
| Program compiles and executes without any runtime, syntax, or logic errors | 0 | 3 |
| The zipped project folder that includes the C++ .cpp source files and screens shots of the code and console and raptor program is uploaded to drop box. | 0 | 1 |
| **Total Points Possible** | 0 | 20 |