

CIS 1068 Assignment 4

Grades

Due: Tuesday, February 8

40 points

Description

This assignment will provide you with practice using Java variables and expressions, `if/else` statements, `Scanner` objects for interactive programming, and writing methods.

Users will provide some information about their performance during the semester, and the program will output the students' weighted semester average according to some rules that follow.

The Program

You will write a Java class called `Grades` that must be saved into a file called `Grades.java`, which should be submitted through Canvas.

As a reminder, you will be using a `Scanner` object for console input, so you will need to `import java.util.Scanner;` into your program.

Your program will read in grades for a student in a fictional course, and it will calculate the overall course grade for the student. Below is one example log of execution of the program (user input is underlined for the sake of clarity in the log --- you do not need to duplicate this behavior).

an example run of the program

```
This program accepts your homework and two exam scores as input and computes your grade in the course.

Homework weight? 50

Exam 1 weight? 20

Using weights of 50 20 30

Homework:

Number of assignments? 10

Average Homework grade? 8.45

Number of late days used? 2

Labs attended? 4

Total points = 100.5 / 140

Weighted score = 35.89

Exam 1:

Score? 81

Curve? 0

Total points = 81 / 100

Weighted score = 16.2
```

Exam 2:

Score? 95Curve? 10

Total points = 100 / 100

Weighted score = 30.0

Course grade = 82.09

The course grade is a weighted average of three components: a homework grade, an exam 1 grade, and an exam 2 grade. To compute a weighted average, the student's point scores for each component are divided by the total points for that component and multiplied by that component's weight.

The homework score is determined by the student's average score on a series of assignments, as well as points for attending labs. Homeworks are worth 10 points, and each lab is worth 4 points, and the number of homeworks and labs is the same. The number of assignments is entered by the user, as is the student's average homework score and the number of labs attended, as shown in the execution log. If a student hands an assignment in late, he or she uses "late days." For this course, if the student uses more late days than half the number of assignments, the assignment grade should be reduced by 10%. If the student uses no late days, the student should get 5 extra credit points for the homeworks. However, a student cannot earn more points than the maximum available for homeworks; a student that averages 10 points per assignment and never hands any in late has already earned the maximum number of homework points and will get no extra credit for being punctual.

Exams are worth 100 points each, and potentially are curved by adding some *curve points* to every students' grade. However, students cannot earn more than 100 points on an exam.

Your program should be careful to correct bad numeric values entered by the user. If a user enters a negative number or zero for the number of assignments, the student should receive full credit for the homework grade. If a user enters a negative number for the average homework grade or an exam grade, it should be treated as a zero. If a user enters a higher score than the maximum number of points available for homeworks or an exam, the score should be capped.

In the log of execution shown, the course has 50% weight for homework, 20% weight for exam 1, and 30% weight for exam 2. There are 10 homework assignments. The student attended 4 labs (earning 16 points for doing so). The student received an exam 1 score of 81. The student earned an exam 2 score of 95; the exam was curved by +10 points, but exam scores are capped at 100, so the student was given 100 for exam 2.

The following calculations produce the student's course grade from the above log of execution:

```
grade = weightedHomeworkScore + weightedExam1Score + weightedExam2Score
weightedHomeworkScore = 50 * (8.45 * 10 + 4 * 4) / (10 * 10 + 4 * 10)
weightedExam1Score = 20 * 81 / 100
weightedExam2Score = 30 * 100 / 100
```

Note that the preceding equations are not Java math. In Java, an integer expression such as 81/100 would evaluate to 0, but above the value intended is 0.81.

When the program asks the user for the weights of the three components, it only asks for weights for homework and exam 1. Using this information, the program can deduce the weight of exam 2 as 100 minus the other two weights.

Stylistic Guidelines

When handling numeric data, you are expected to choose appropriately between types `int` and `double`.

Please indent your code and use whitespace to make your program readable. Give meaningful names to variables in your code. Follow Java's naming and capitalization standards.

Include a comment at the beginning of your program with basic information and a description of the program.

rubric

- **[2 points]** Scanner set up properly
- **[2 points]** Readable indentation and spacing
- **[2 points]** Each input read from user has a prompt
- **[3 points]** Proper handling of invalid inputs
- **[15 points]** At least 3 functions to handle various calculations, with arguments passed and a value returned.
- **[5 points]** No use of global variables
- **[5 points]** Descriptive variable and constant names, e.g., exam1Score, EXAM1_WEIGHT
- **[6 points]** Calculates correct semester average