CS201 – Spring 2015-2016 - Sabancı University

Homework #2: Grade Calculator

Due March 2, Wednesday, 19:00 (Sharp Deadline)

Introduction

The aim of this homework is to practice on parametric functions and if statements. The use of if statement is due to the nature of the problem (that is, you cannot finish this homework without using if statement). However, the use of functions is somehow enforced. That means it is possible to accomplish this homework without using functions, but it is a must to use them. The details about the use of functions in this homework are given later in this document.

Description

In this homework, you will write a C++ program that calculates the letter grade for our course CS201. Student name, midterm grades, final grade and average of the homework grades will be your inputs.

The program starts with displaying a welcome message that explains your program. Next phase is input entry for two separate students.

User will first input the name of the student. Afterwards the user must input three separate grades for two midterms and a final. Then the user will be prompted to enter the average of homework grades. As expected the students can't get negative points in the exams, but there are no bonuses either. However, if students submit plagiarized homework's, they will be punished by getting -100 for that homework. Once the inputs are completed, your program will first check if all inputs entered are valid according to given specifications. If all values are correctly entered, then your program will calculate the grade of that student according to the rules of the course, which will be explained in detail in the next section. Finally, your program must output an appropriate message stating the students name and his/her grade and, if necessary, the reason for any invalid inputs.

The chain of inputs, calculations and outputs will be done twice. In other words users will enter two different names, and see both their letter grades. Whenever an input is determined to be erroneous, your program should show an output stating the problem. Please keep in mind that you need to follow the exact steps for a second student. However, if the program fails the input check, it should give out an error message and stop calculation for that particular student (so if inputs for the first student are invalid, you should still be able to get inputs for the second student and do calculations).

As mentioned above you cannot finish the program without using if statements. In addition to that we enforce that you write parametric functions. As a coding convention if a segment of code is repeating itself more than once, then it is best to make it into a function to avoid any code duplication.

VERY IMPORTANT!

Your programs will be compiled, executed and evaluated automatically; therefore you should definitely follow the rules for inputs and outputs. See **Sample Runs** section for some examples.

- **Order of inputs and outputs** must be in the mentioned format.
- **Prompts before inputs and outputs** must be **exactly the same text** with examples.

Following these rules is crucial for grading, otherwise our software will not be able to process your outputs and you will lose some grades in the best scenario.

Input Check

In your first homework, you were not expected to do any input checks, but in this homework you **have to** perform input check as detailed below:

- Exam grades are non-negative integers with 100 being the highest grade and 0 being the lowest grade
- Average homework grade is a real number between -100 and 100 inclusive

If there are more than one input check errors, it does not matter which one is displayed first. You may display only the first invalid input or you may display all input errors in any order.

However, you are NOT expected to (and please DO NOT try to) re-input variables if erroneous inputs are entered. Such functionality requires loops or recursion that we have not covered yet. On the other hand, you are going to pass some of the inputs and calculated values to these functions as parameters. These functions and parameter passing will be explained in the subsequent section.

IMPORTANT!

If your code does not compile, you will get **zero**. Please be careful about this and double check your code before submission.

Calculations

The grade calculation rules for this version of CS201 is as follows:

1. Exams can be [0,100]
2. Homework average grade can be [-100,100]
3. Your midterm grades must add up to 40 (mt1+mt2) to pass the course, otherwise the letter grade will be F.
4. Your final grade must be at least 20 to pass the course, otherwise the letter grade will be F.
5. The distribution is: MT1(25%), MT2(25%), F(35%), HW(15%)
6. The exam average must be at least 35 to pass the course, otherwise the letter grade will be F.
7. If homework average is larger than the double of the exam average, then the grade will be lowered by a letter grade.

Description	Formula
Course Average (Rule 5)	(mt1 + mt2) * 0.25 + f * 0.35 + hw * 0.15
Exam Average (Rule 6)	((mt1 + mt2) * 0.25 + f * 0.35) / 0.85
Grade Letter Decreaser (Rule 7)	HW_average > 2 * Exam_Average

Letter	Points
A	>90
A-	>85
B+	>80
B	>75
B-	>70
C+	>60
C	>55
C-	>50
D+	>45
D	>40

Use of Functions (EXTREMELY IMPORTANT!)

You have to follow the specifications below for function declaration and callings. The grading criteria will include proper use of these parametric functions. Do NOT use any global variables (variables defined outside the functions) to avoid parameter use. Unnecessary code duplication will cause grade reduction as well.

In the first homework you were not supposed to implement any functions. However, in this homework you are expected to (actually you have to) use some function(s). The guidance about using functions in this homework is below.

A total of four user-defined functions (other than main) must be implemented. You have to implement and use these four functions. If you don't, your grade will be lowered because of the missing functions. On top of these functions, you may use other functions if you want.

Function1: This function will be used to prompt for the inputs and to get them from the user. After all the inputs are entered, it will call Function2 with the inputted parameters.

Function2: This function will take all the input parameters and check the correctness of the necessary ones. If there are no erroneous inputs, it will call Function3. Otherwise, it will print out appropriate error messages separately. Hint1: Use if and else-if statements. Hint2: Check rules (1, 2).

Function3: This function will take all the valid input parameters and check for Rule 3, 4, 6 and 7. It will also calculate the total points (Rule 5). Once the calculations are finished, it will call Function4.

Function4: This function will calculate the letter grade according to the total point. So, you have to pass some of the variables from Function3 to Function4.

Other than these four functions, you will also have the *main* function. The welcome message display will be in *main*. Moreover, you will call Function1 from *main* twice for the two different students.

Needless to say, you have to name these for functions using **meaningful** identifiers, not as Function1, Function2, etc.

No abrupt program termination please!

You may want to stop the execution of the program at a specific place in the program. Although there are ways of doing this in C++, it is not a good programming practice to abruptly stop the execution in the middle of the program. Therefore, your program flow should continue until the end of the main function and finish there.

Sample Runs

Below, we provide some sample runs of the program that you will develop. The *italic* and **bold** phrases are inputs taken from the user. The introductory line (This program bla bla bla) is intentionally left incomplete. You are expected to use your imagination and creativity there to introduce your program. However, as mentioned above, the input and the output order and their prompts should remain intact.

Sample Run 1

```
This program bla bla bla...
Please enter your name and surname: Slade Wilson
Slade Wilson, please enter your midterm and final grades: 56 68 64
Slade Wilson, please enter the average score of your homeworks: 89.6

Slade Wilson, you get C+

Please enter your name and surname: Wade Wilson
Wade Wilson, please enter your midterm and final grades: 28 28 29
Wade Wilson, please enter the average score of your homeworks: 85

Wade Wilson, your lack of average grades disturbs me: You get F
```

Sample Run 2

```
This program bla bla bla...
Please enter your name and surname: Ash Ketchum
Ash Ketchum, please enter your midterm and final grades: 19 17 45
Ash Ketchum, please enter the average score of your homeworks: 75.5

Ash Ketchum, your lack of grades in your midterms disturbs me: You get F

Please enter your name and surname: Gary Oak
Gary Oak, please enter your midterm and final grades: 50 41 19
Gary Oak, please enter the average score of your homeworks: 80.2

Gary Oak, your lack of point in your final disturbs me: You get F
```

Sample Run 3

```
This program bla bla bla...
Please enter your name and surname: Michael Cera
Michael Cera, please enter your midterm and final grades: -2 100 98
Michael Cera, please enter the average score of your homeworks: 100

Michael Cera, your exam grades cannot be below 0, we aren't that harsh

Please enter your name and surname: Mary Elizabeth
Mary Elizabeth, please enter your midterm and final grades: 49 49 47
Mary Elizabeth, please enter the average score of your homeworks: 98.8

Mary Elizabeth, unfortunately your grade will be lowered
Mary Elizabeth, you get C-
```

Sample Run 4

```
This program bla bla bla...
Please enter your name and surname: Oliver Queen
Oliver Queen, please enter your midterm and final grades: 140 140 140
Oliver Queen, please enter the average score of your homeworks: 140

Oliver Queen, over 100 point in an exam? We don't give bonuses, we aren't
that nice either

Please enter your name and surname: Barry Allen
Barry Allen, please enter your midterm and final grades: 35 60 70
Barry Allen, please enter the average score of your homeworks: -18.8

Barry Allen, you get D+
```

Sample Run 5

```
This program bla bla bla...
Please enter your name and surname: Rick Sanchez
Rick Sanchez, please enter your midterm and final grades: 38 38 40
Rick Sanchez, please enter the average score of your homeworks: 35

Rick Sanchez, you failed this course: F

Please enter your name and surname: Morty Smith
Morty Smith, please enter your midterm and final grades: 94 100 97
Morty Smith, please enter the average score of your homeworks: 98.9

Morty Smith, you get A
```

Sample Run 6

```
This program bla bla bla...
Please enter your name and surname: Kyle Rayner
Rick Sanchez, please enter your midterm and final grades: 57 35 63
Rick Sanchez, please enter the average score of your homeworks: 76.2

Rick Sanchez, you get C

Please enter your name and surname: Hal Jordan
Morty Smith, please enter your midterm and final grades: 74 82 71
Morty Smith, please enter the average score of your homeworks: 85.6

Morty Smith, you get B
```

General Rules and Guidelines about Homeworks

The following rules and guidelines will be applicable to all homeworks, unless otherwise noted.

How to get help?

You may ask questions to TAs (Teaching Assistants) of CS201. Office hours of TAs are at the class website. Recitations will partially be dedicated to clarify the issues related to homework, so it is to your benefit to attend recitations.

Moreover, in the recitations for the first homework, there will be a demonstration on how to prepare homework for submission and how to submit it to SUCourse. This process is not so straightforward. We have experienced several unsuccessful submissions in the previous years. Therefore, we strongly recommend you attend this demo.

What and Where to Submit

Please see the detailed instructions below/in the next page. The submission steps will get natural/easy for later homeworks.

Grading and Objections

Careful about the semi-automatic grading: Your programs will be graded using a semi-automated system. Therefore you should follow the guidelines about input and output order. Otherwise semi-automated grading process will fail for your homework, and you may get a zero, or in the best scenario you will lose points.

Grading:

- Late penalty is 10% off of the full grade and only one late day is allowed.
- **Having a correct program is necessary, but not sufficient to get the full grade. Comments, indentation, meaningful and understandable identifier names, informative introduction and prompts, and especially proper use of required functions, unnecessarily long program (which is bad) and unnecessary code duplications (which is also bad) will also affect your grade.**
- Please submit your own work only (even if it is not working). It is really easy to find out "similar" programs!
- For detailed rules and course policy on plagiarism, please check out http://myweb.sabanciuniv.edu/gulsend/su_current_courses/cs-201-spring-2008/plagiarism/ and keep in mind that

Plagiarism will not be tolerated!

Grade announcements: Grades will be posted in SUCourse, and you will get an Announcement at the same time. You will find the grading policy and test cases in that announcement.

Grade objections: It is your right to object to your grade if you think there is a problem, but before making an objection please try the steps below and if you still think there is a problem, contact the TA that graded your homework from the email address provided in the comment section of your announced homework grade or attend the specified objection hour in your grade announcement.

- Check the comment section in the homework tab to see the problem with your homework.
- Download the .zip file you submitted to SUCourse and try to compile it.
- Check the test cases in the announcement and try them with your code.
- Compare your results with the given results in the announcement.

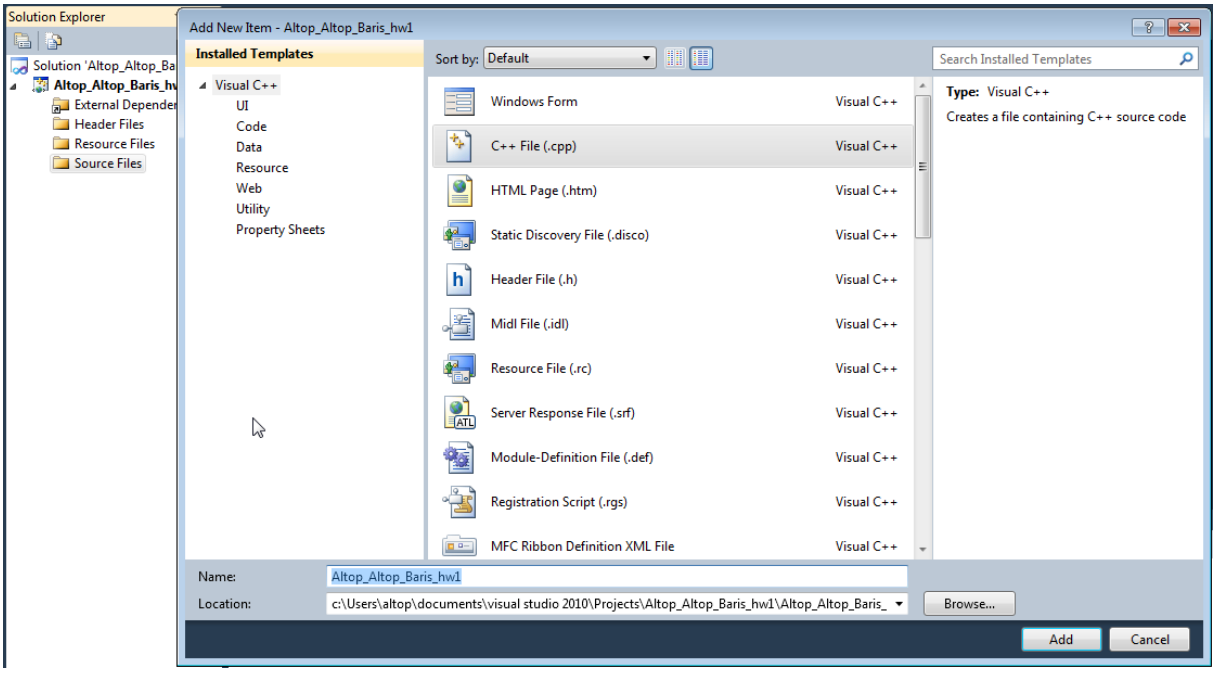
What and where to submit (IMPORTANT)

Submissions guidelines are below. Most parts of the grading process are automatic. Students are expected to strictly follow these guidelines in order to have a smooth grading process. If you do not follow these guidelines, depending on the severity of the problem created during the grading process, 5 or more penalty points are to be deducted from the grade.

Add your name to the program: It is a good practice to write your name and last name somewhere in the beginning program (as a comment line of course).

Name your submission file:

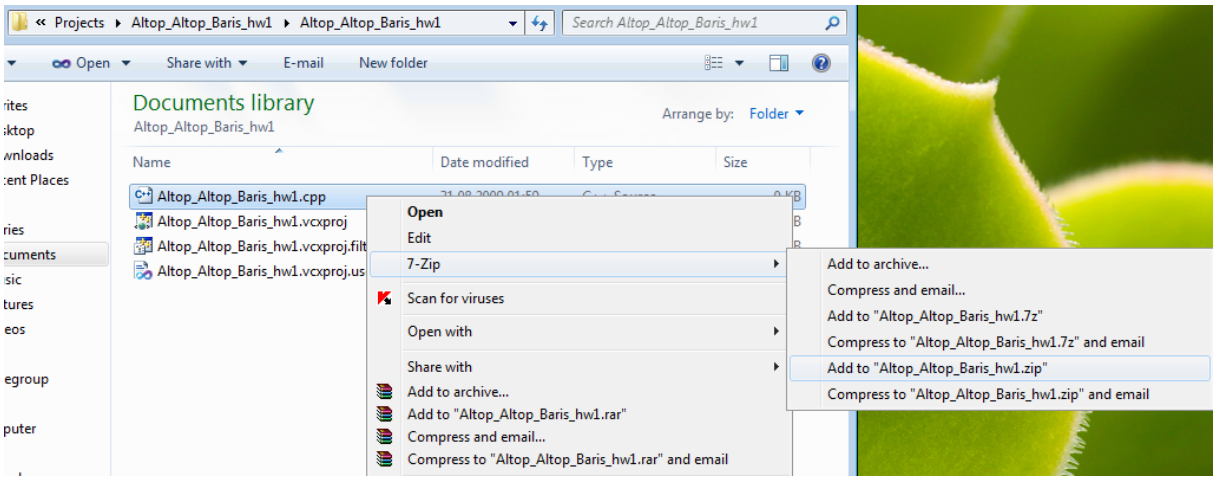
- ❑ Use only English alphabet letters, digits or underscore in the file names. Do not use blank, Turkish characters or any other special symbols or characters.
- ❑ Name your cpp file that contains your program as follows.
  "**SUCourseUserName_YourLastname_YourName_HWnumber.cpp**"



- Your SUCourse user name is actually your SUNet user name which is used for checking sabanciuniv e-mails. Do NOT use any spaces, non-ASCII and Turkish characters in the file name. For example, if your SUCourse user name is cago, name is Çağlayan, and last name is Özbugsızkodyazaroğlu, then the file name must be:

Cago_Ozbugsizkodyazaroglu_Caglayan_hw2.cpp

- Do not add any other character or phrase to the file name.
- Make sure that this file is the latest version of your homework program.
- Compress this cpp file using WINZIP or WINRAR programs. Please use "**zip**" compression. "rar" or another compression mechanism is NOT allowed. Our homework processing system works only with zip files. Therefore, make sure that the resulting compressed file has a zip extension.



- Check that your compressed file opens up correctly and it contains your **cpp** file. You will receive no credits if your compressed zip file does not expand or it does not contain the correct file.
- The naming convention of the zip file is the same as the cpp file (except the extension of the file of course). The name of the zip file should be as follows.

"**SUCourseUserName_YourLastname_YourName_HWnumber.zip**"

For example zubzipler_Zipleroglu_Zubeyir_hw2.zip is a valid name, but hw2_hoz_HasanOz.zip, HasanOzHoz.zip are NOT valid names.

<u>Submission:</u>

- Submit via SUCourse ONLY! You will receive no credits if you submit by other means (e-mail, paper, etc.).
  1) Click on "Assignments" at CS201 SUCourse (not the CS201 web site).
  2) Click Homework 2 in the assignments list.
  3) Click on "Add Attachments" button.
  4) Click on "Browse" button and select the zip file that you generated.
  5) Now, you have to see your zip file in the "Items to attach" list.
  6) Click on "Continue" button.
  7) Click on "Submit" button. We cannot see your homework if you do not perform this step even if you upload your file.

Resubmission:

- After submission, you will be able to take your homework back and resubmit. In order to resubmit, follow the following steps.
    1) Click on "Assignments" at CS201 SUCourse.
    2) Click Homework 2 in the assignments list.
    3) Click on "Re-submit" button.
    4) Click on "Add/remove Attachments" button
    5) Remove the existing zip file by clicking on "remove" link. This step is very important. If you do not delete the old zip file, we receive both files and the old one may be graded.
    6) Click on "Browse" button and select the new zip file that you want to resubmit.
    7) Now, you have to see your new zip file in the "Items to attach" list.
    8) Click on "Continue" button.
    9) Click on "Submit" button. We cannot see your homework if you do not perform this step even if you upload your file.

Successful submission is one of the requirements of the homework. If, for some reason, you cannot successfully submit your homework and we cannot grade it, your grade will be 0.

Good Luck!
Ece Egemen and Gülşen Demiröz