

**CS201 – SPRING 2020-2021**  
**Take-Home Exam 0**  
**– CS201 Course Numeric Grade Calculation –**  
**Due March 10<sup>th</sup>, Wednesday, 23:55 (Sharp Deadline)**

## Introduction

The aim of this take-home exam is to practice on basics of C++ (input, output, assignment, arithmetic, etc.) and decision making (conditional if statements). The use of if-statements is due to the nature of the problem; that is, you cannot finish this take-home exam without using decision making.

## Description

In CS201, weights of the grading items on the course numeric grade are as given in Table 1. However, the weighted average is not the only criterion in letter grading, and thus, in the calculation of the course numeric grade.

Table1. Tentative Grading for CS201

GRADING ITEM	PERCENTAGE
Take-Home Exams	20%
Quizzes	20%
Midterm Exam	30%
Final Exam	30%

Here are some of the rules, which will be used in this take-home exam:

- If you miss the final exam, and if you do not take the make-up exam for that missing exam; then you will directly get an F grade.
  - If the entered grade is -1, it means the student has missed that exam.
- If one of your midterm or final exams is plagiarized (i.e. grade is -100); then you will directly get an F grade.
- If your exams' weighted average is below 30, you will fail the course even if your total grade is equal or above the overall passing grade.
- All of the assigned take-home exams and quizzes will be taken into consideration in the course numeric grade.

- For the first time, all plagiarized take-home exam or quiz owners receive - 100 (minus hundred).
  - Second time, the student fails the course automatically.

In this take-home exam, you will implement a C++ program that calculates the course numeric grade of a CS201 student. The inputs to your program will be (i) name of the student, (ii) midterm and final exam grades, (iii) take-home exam grades, and (iv) quiz grades of the student. Your program will then calculate the weighted exam average, take-home exam average, quiz average, and the course numeric grade, based on some conditions as described below.

During designing the algorithm for your program, you will follow the below listed assumptions in this take-home exam:

- There is no make-up exam for the regular midterm and final exams.
- There are a total of 5 (five) take-home exams, and the student has submitted a work for each of those take-home exams
  - THE-1: 3%
  - THE-2, THE-3, THE-4: 4% (each)
  - THE-5: 5%
- There are a total of 4 (four) quizzes, and the student has attended each of those quizzes. All quizzes have equal weight (5%).

You may implement your algorithm all in the *main* function, i.e. you are not expected to write any user-defined functions, but of course you can write some, if you prefer to do so.

Your take-home exams will be automatically graded using GradeChecker, so it is very important to satisfy the exact same output given in the sample runs. You can utilize GradeChecker (<http://learnt.sabanciuniv.edu/GradeChecker/>) to check whether your implementation is working in the expected way. To be able to use GradeChecker, you should upload all of your files used in the take-home exam (**only** *your\_main\_cpp* file for this take-home exam). Additionally, you should submit all of your files to SUCourse (**only** *your\_main\_cpp* file for this take-home exam) **without zipping** them. **Just a reminder, you will see a character ¶ which refers to a newline in your expected output.**

**The name of your main source (cpp) file should be in the expected format:** "SUCourseUsername\_THEnumber.cpp" (all lowercase letters). Please check the submission procedures of the take-home exam, which are listed at the end of this document.

## Inputs, Flow of the Program and Outputs

The inputs of the program and their order are explained below. It is extremely important to follow this order with the same characters since we automatically process your programs. ***Thus, your work will be graded as 0 unless the order is entirely correct.*** Please see the "Sample Runs" section for some examples.

The prompts of the input statements to be used has to be exactly the same as the prompts of the "Sample Runs".

At the beginning of your program, the user will be prompted to enter his/her name as the first input. The name of the user should be displayed whenever appropriate in the prompts and in the output text. Please see the "Sample Runs" section for some examples.

After the user enters his/her name, (s)he will be asked to provide his/her midterm and final exam grades. Once inputted, your program will first check if either one of the grades of the exams is **-100** or not. After that your program will check if either of the grades of the exams is **-1** (did not enter the exam) or not.

If all grades are positive, then your program should calculate the weighted exam average as given below:

$$\text{weighted\_exam\_avr} = (0.30 \times \text{midterm\_grade} + 0.30 \times \text{final\_grade}) / 0.6$$

If the weighted exam average of the user is below 30, (s)he will fail the course due to the course rules given before. In such a case, your program will display an appropriate message, and then, your program will finish without even getting or prompting for the other inputs.

On the other hand, if the weighted exam average of the user is equal to or greater than 30, then your program will continue. This time, the user will be asked to enter the grades of his/her 5 take-home exams. Once inputted, your program will calculate the number of THE grades that are plagiarized (-100). If 2 or more are -100, then the student will fail the course automatically. In such a case, your program will display an appropriate message, and then, your program will finish without even getting or prompting for the other inputs.

If the count is below two then your program will calculate the weighted average of those take-home exams and compare it with the average of the midterm and final exams.

Contribution of the take-home exams to the overall grade will be calculated according to the formula given below:

$$\begin{aligned}
 \text{take\_home\_grade} &= \begin{cases} \text{take\_home\_avr} & \text{if } \text{ratio} \leq 2 \\ \text{take\_home\_avr} \times (3 - \text{ratio}) & \text{if } 2 < \text{ratio} < 3 \\ 0 & \text{if } \text{ratio} \geq 3 \end{cases} \\
 \text{ratio} &= \text{submitted\_take\_home\_avr} / \text{weighted\_exam\_avr} \\
 \text{weighted\_exam\_avr} &= (0.3 \times \text{midterm\_grade} + 0.3 \times \text{final\_grade}) / 0.6 \\
 \text{course\_numeric\_grade} &= \text{quiz\_grade} \times 0.2 + \text{take\_home\_grade} \times 0.2 + \\
 &\quad \text{midterm\_grade} \times 0.3 + \text{final\_exam\_grade} \times 0.3
 \end{aligned}$$

If the take home average is below zero, this ratio rule is not applied (take home grade is equal to take home average).

Then, your program will prompt for the quiz grades of the user. The user will input 4 quiz grades. Once inputted, your program will calculate the number of quiz grades that are plagiarized (-100). If 2 or more are -100, then the student will fail the course automatically. In such a case, your program will display an appropriate message, and then, your program will finish without even getting or prompting for the other inputs.

Thereafter, your program will also check if the student has 1 THE and 1 Quiz plagiarized (-100). If in total the student has 2 grades -100, the student will fail the course automatically. In such a case, your program will display an appropriate message, and then, your program will finish without even calculating the final numerical grade.

If none apply your program will calculate the average of the quiz grades and compare it with the weighted average of the midterm and final exams. Contribution of the quizzes to the overall grade will be calculated according to the formula given below:

$$\begin{aligned}
 \text{quiz\_grade} &= \begin{cases} \text{quiz\_avr} & \text{if } \text{ratio} \leq 2 \\ \text{quiz\_avr} \times (3 - \text{ratio}) & \text{if } 2 < \text{ratio} < 3 \\ 0 & \text{if } \text{ratio} \geq 3 \end{cases} \\
 \text{ratio} &= \text{submitted\_quiz\_avr} / \text{weighted\_exam\_avr} \\
 \text{weighted\_exam\_avr} &= (0.3 \times \text{midterm\_grade} + 0.3 \times \text{final\_grade}) / 0.6 \\
 \text{course\_numeric\_grade} &= \text{quiz\_grade} \times 0.2 + \text{take\_home\_grade} \times 0.2 + \\
 &\quad \text{midterm\_grade} \times 0.3 + \text{final\_exam\_grade} \times 0.3
 \end{aligned}$$

If the quiz average is below zero, this ratio rule is not applied (quiz grade is equal to quiz average).

Before calculating the final numerical grade, your program should display the weighted exam average, THE average with ratio-applied THE average and Quiz average with ratio-applied quiz average each on separate lines.

Lastly, your program will calculate and display the course numeric grade of the user, which is to be calculated as given below:

$$\begin{aligned} \text{course\_numeric\_grade} &= \text{quiz\_grade} * 0.2 \\ &+ \text{take\_home\_grade} * 0.2 \\ &+ \text{midterm\_exam\_grade} * 0.3 \\ &+ \text{final\_exam\_grade} * 0.3 \end{aligned}$$

You may assume that the user will always enter either positive values or -100 (minus hundred) correctly for all inputs. Therefore, you do not need to perform any kind of input checks.

Please refer to the "Sample Runs" section for some examples and further details.

### **IMPORTANT!**

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

### **VERY IMPORTANT!**

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

- **Order of inputs and outputs** must be in the mentioned format.

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

## **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. You have to display the required information in the same order and with the same words and characters as below.

### ***Sample Run 1***

The purpose of this program is to calculate your CS201 grade.  
What is your name? ***Gulsen***  
Hello Gulsen, what are your midterm and final exam grades? ***20 30***  
Gulsen, your weighted exam average is less than 30, you will fail the course!

### ***Sample Run 2***

The purpose of this program is to calculate your CS201 grade.  
What is your name? ***Baris***  
Hello Baris, what are your midterm and final exam grades? ***83 -100***  
Baris, one of your midterm or final exam grade is -100, you will fail the course!

### ***Sample Run 3***

The purpose of this program is to calculate your CS201 grade.  
What is your name? ***Sevval***  
Hello Sevval, what are your midterm and final exam grades? ***-1 77***  
Sevval, you did not enter the midterm exam, you will fail the course!

### ***Sample Run 4***

The purpose of this program is to calculate your CS201 grade.  
What is your name? ***Nasim***  
Hello Nasim, what are your midterm and final exam grades? ***68 -1***  
Nasim, you did not enter the final exam, you will fail the course!

### **Sample Run 5**

The purpose of this program is to calculate your CS201 grade.  
What is your name? **Furkan**  
Hello Furkan, what are your midterm and final exam grades? **45 53**  
What are the grades of your 5 take-home exams? **90 58 -100 0 -100**  
Furkan, at least two of your THE grades are -100, you will fail the course!

### **Sample Run 6**

The purpose of this program is to calculate your CS201 grade.  
What is your name? **Mohsin**  
Hello Mohsin, what are your midterm and final exam grades? **67 90**  
What are the grades of your 5 take-home exams? **100 88 -100 80 60**  
What are the grades of your 4 quizzes? **85 70 -100 30**  
Mohsin, one of your THE and one of your quiz grades is -100, you will fail the course!

### **Sample Run 7**

The purpose of this program is to calculate your CS201 grade.  
What is your name? **Berfin**  
Hello Berfin, what are your midterm and final exam grades? **90 82**  
What are the grades of your 5 take-home exams? **100 100 80 90 70**  
What are the grades of your 4 quizzes? **90 85 80 95**  
Berfin, your weighted exam average is: 86  
Berfin, your THE average is: 86.5 and you will get 86.5 of it.  
Berfin, your quiz average is: 87.5 and you will get 87.5 of it.  
Berfin, your grade for CS201 is: 86.4

### **Sample Run 8**

The purpose of this program is to calculate your CS201 grade.  
What is your name? **Ufuk**  
Hello Ufuk, what are your midterm and final exam grades? **33 29**  
What are the grades of your 5 take-home exams? **100 100 70 80 100**  
What are the grades of your 4 quizzes? **60 80 70 70**  
Ufuk, your weighted exam average is: 31  
Ufuk, your THE average is: 90 and you will get 8.70968 of it.  
Ufuk, your quiz average is: 70 and you will get 51.9355 of it.  
Ufuk, your grade for CS201 is: 30.729

### **Sample Run 9**

The purpose of this program is to calculate your CS201 grade.  
What is your name? **Ekberjan**  
Hello Ekberjan, what are your midterm and final exam grades? **37 33**  
What are the grades of your 5 take-home exams? **100 100 90 100 100**  
What are the grades of your 4 quizzes? **60 55 50 50**  
Ekberjan, your weighted exam average is: 35  
Ekberjan, your THE average is: 98 and you will get 19.6 of it.  
Ekberjan, your quiz average is: 53.75 and you will get 53.75 of it.  
Ekberjan, your grade for CS201 is: 35.67

### **Sample Run 10**

The purpose of this program is to calculate your CS201 grade.  
What is your name? **Mert**  
Hello Mert, what are your midterm and final exam grades? **30 30**  
What are the grades of your 5 take-home exams? **100 28 37 60 40**  
What are the grades of your 4 quizzes? **100 90 100 100**  
Mert, your weighted exam average is: 30  
Mert, your THE average is: 50 and you will get 50 of it.  
Mert, your quiz average is: 97.5 and you will get 0 of it.  
Mert, your grade for CS201 is: 28

### **Sample Run 11**

The purpose of this program is to calculate your CS201 grade.  
What is your name? **Deniz**  
Hello Deniz, what are your midterm and final exam grades? **80 68**  
What are the grades of your 5 take-home exams? **83 67 90 80 50**  
What are the grades of your 4 quizzes? **60 -100 -100 90**  
Deniz, at least two of your quiz grades are -100, you will fail the course!



## General Rules and Guidelines about Homeworks

The following rules and guidelines will be applicable to all take-home exams, unless otherwise noted.

### - How to get help?

You can use GradeChecker (<http://learnt.sabanciuniv.edu/GradeChecker/>) to check your expected grade. Just a reminder, you will see a character ¶ which refers to a newline in your expected output.

You may ask questions to TAs (Teaching Assistants) or LAs (Learning Assistants) of CS201. Office hours of TAs/LAs are at the [course website](#).

### - What and Where to Submit

You should prepare (or at least test) your program using MS Visual Studio 2012 C++ (Windows users) or using Xcode (macOS users).

It'd be a good idea to write your name and last name in the program (as a comment line of course). Do not use any Turkish characters anywhere in your code (not even in comment parts). If your name and last name is "İnanç Arın", and if you want to write it as comment; then you must type it as follows:

*// Baris Altop*

Submission guidelines are below. Since the grading process will be automatic, students are expected to strictly follow these guidelines. If you do not follow these guidelines, your grade will be 0.

- Name your submission file as follows:
  - 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\_THEnumber.cpp"**
  - Your SUCourse user name is actually your SUNet username, which is used for checking sabanciuniv emails. Do NOT use any spaces, non-ASCII and Turkish characters in the file name (**use only lowercase letters**). For example, if your SUCourse username is "**altop**", then the file name should be: **altop\_the0.cpp** (please only use lowercase letters).
  - Do not add any other character or phrase to the file name.

- Please make sure that this file is the latest version of your take-home exam program.
- Submit your work **through SUCourse only**! You can use GradeChecker only to see if your program can produce the correct outputs both in the correct order and in the correct format. It will not be considered as the official submission. You must submit your work to SUCourse. You will receive no credits if you submit by any other means (email, paper, etc.).
- If you would like to resubmit your work, you should first remove the existing file(s). This step is very important. If you do not delete the old file(s), we will receive both files and the old one may be graded.

### - **Grading, Review and Objections**

Be careful about the automatic grading: Your programs will be graded using an automated system. Therefore, you should follow the guidelines on the input and output order. Moreover, you should also use the same text as given in the "Sample Runs" section. Otherwise, the automated grading process will fail for your take-home exam, and you may get a zero, or in the best scenario, you will lose points.

#### Grading:

- There is NO late submission. You need to submit your take-home exam before the deadline. Please be careful that SUCourse time and your computer time may have 1-2 minutes differences. You need to take this time difference into consideration.
- Successful submission is one of the requirements of the take-home exam. If, for some reason, you cannot successfully submit your take-home exam and we cannot grade it, your grade will be 0.
- If your code does not work because of a syntax error, then we cannot grade it; and thus, your grade will be 0.
- Please submit your **own** work only. It is really easy to find "similar" programs!
- Plagiarism will not be tolerated. Please check our plagiarism policy given in the [Syllabus](#) or on the [course website](#).

**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 take-home exam from the email address provided in the comment section of your announced take-home exam grade or attend the specified objection hour in your grade announcement.

- Check the comment section in the take-home exam tab to see the problem with your take-home exam.
- Download the 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.

***Good Luck!***

***Şevval Şimşek & Gülşen Demiröz & Barış Altop***