

## CSc 110 Lab Assignment 4

### Calculate your Grade

Due June 24 @ 11:55pm

**Learning Outcomes:** When you have completed this assignment, you should understand:

1. How to implement conditional statements (if-statements and if/else-statements)
2. How to analyze the flow of choices to ensure correct program logic.
3. How to design a suite of tests that will check the various cases in a branching program (i.e. one that contains conditional statements).
4. How to use parameterized loops within methods.
5. How to design and use indefinite loops to handle invalid inputs.

### Motivation

In this assignment you are given the opportunity to experiment with conditional statements and nested loops and continue to develop your skills in deciding how to 'method-ize' a program. There are many choices to be made in the development of this code, different programmers will make different choices. Like two different works of art, there is more than one way to make a beautiful program. Taking an opportunity to explore alternative beautiful programs is an excellent learning experience. This assignment also represents an opportunity to examine test strategy. The various inputs can all be tested for correctness. Try to consider all the possible incorrect inputs.

### Details

The problem requires the creation of an automated CSc 110 grade calculation program. The program is interactive and prompts the user for all the scores for one student. The result is the percentage and letter grade for that student.

For the CSc 110 course, the course outline shows that grades are calculated as follows:

Coursework	Weight (out of 100%)	
Assignments	24%	
In-Class	6%	
Labs	10%	1% awarded in each of 10 lab classes
Midterm Exams	20%	
Final Exam	40%	

All assignments are marked out of a maximum of 10 marks each. The lab attendances will be added together and input as a single number (out of a maximum of 10). The midterm and final are each marked out of a maximum of 100 marks.

Using the grades for each component and the weightings above a total course mark (maximum possible: 100), the final percentage, is converted to a letter grade according to the following scale:

F	D	C	C+	B-	B	B+	A-	A	A+
0-49	50-59	60-65	66-71	72-77	78-82	83-87	88-92	93-96	97-100

F	D	C	C+	B-	B	B+	A-	A	A+
0-49.4	49.5-59.4	59.5-65.4	65.5-71.4	71.5-77.4	77.5-82.4	82.5-87.4	87.5-92.4	92.5-96.4	96.5-100

In order to pass the course, a student must obtain a passing grade on the final exam. Thus, a student who receives an overall final grade greater than or equal to 49.5% but fails the final exam will receive an F.

Exactly what constitutes a 'passing grade' on the final exam is determined by the instructor at the end of the term. (Although it is likely that the instructor chooses 50% to be the passing grade, (s)he may determine, for example, that the final exam was particularly difficult and decide that the passing grade on the final exam is only 42%!!)

### **Planning Activity: Program Design Worksheet**

Complete Program Design Worksheet (below), using the problem data as described above. Be sure to use proper pseudo-code that is **not** a program. You may discuss your completed worksheet with the instructor in office hours or with consultants.

### **The program:**

The program prompts the user for all of the grades. (Specifically, the user will input each of the grades on the keyboard.) Each input should be a raw score out of the maximum possible for each component (a number out of 20 for each assignment, a number out of 5 for the lab, etc.). In the event a number exceeds the maximum or is less than zero, the user should be prompted to re-enter the value. The determination, by the instructor, of the minimum passing grade on the final exam must also be input.

The program will calculate and output the student's final percentage and letter grade. Sample output for this program is:

## COURSE GRADE CALCULATOR

Author: L. Jackson October 2010 update May 2011

Purpose: Calculated the weighted grade for a student in a course

Inputs: Assignment, Lab, Quiz and Exam grades

Passing Grade for Final ==> 50

Input ID number ==> 12345

Input Assignments (maximum 20) :

#1==>16

#2==>16

#3==>16

#4==>16

#5==>16

#6==>16

#7==>16

#8==>16

Input Lab Attendance(maximum 5) :

#1==>7

INPUT TOO LARGE: Type a number in the range: 0-5

#1 ==>4

Input quiz(maximum 20.0) :

#1 ==>-2

INPUT TOO SMALL: Type a number in the range: 0-20

#1==>16

Input Midterm(maximum 100) :

#1==>80

Input Final Exam(maximum 100) :

#1==>80

12345 Grade = 80.0 Letter = B

One item that the marker will be looking for: the use of a single method that is used over repeatedly, once for each type of activity, say lab, assignment, midterm, etc. The method would input each items of the activity (foe example, each lab or assignment grade or the specific exam grade) then average them and weight them such that the output (returned value) is the weighted average of the activity. This method should be used for activities that have multiple items, such as labs and assignments, or only single items, such as the midterm or final exam.

Once you believe you have a correct program make sure you have tested it carefully for:

- invalid grade values are identified as too high or too low and the user may keep trying
- letter grade boundaries are correct and tested
- weight calculations are correct and tested
- the passing grade for the final is collected from standard input and used correctly

- failure of the final exam will result in an 'F' letter grade, always
- Also, carefully evaluate your use of constants, methods, and control structures to ensure ways that create high-quality software.

Please discuss your code with the consultants and/or the instructional team to find opportunities for improving the quality.

### What to hand in:

1. A electronic copy of the program design worksheet.
2. An electronic copy of the final \*.java program: This is handed in using the assignments link on the course Connex site.

### More to Discover

- Would you trust your computer program to decide on your CSc 110 grade? Would you trust a computer to decide if you have completed all the requirements necessary to be awarded a degree? What about tracking your bank account or making a medical diagnosis? Doctors that are good at making diagnoses often rely on a vast range of knowledge and experience. They are able to sort through the available data to find the relevant pieces of information and then compare that information to patterns that they have seen before. They can also make deductions and inferences to discover (or lead them to inquire about) extra information that may not be obvious. Many of the pattern recognition and inference procedures that doctors use can be expressed using a computer program. Such programs are examples of [artificial intelligence](#). In fact, computers can often recognize more nuanced patterns and perform more complex inferences that a person can, simply because they have larger working memory. However, the one thing that computers lack is common sense. If an aged patient shuffles into a doctor's office with a cane, complaining of ankle pain, the doctor can infer fairly quickly that the pain is probably not due to an acute running injury—a computer on the other hand would have to ask, "Did you hurt yourself running?"
- Computers can be programmed to recognize patterns, but they can also be used to *find* new patterns. The current buzzword for this process is [data mining](#). Companies such as Amazon employ programs to search through vast databases looking for statistical connections between one piece of information and another. Once they have identified relationships between various items, these patterns can then be exploited to produce customized advertisements such as, "You recently bought *x*. If you liked it, you may also be interested in *y*." Google provides similar functionality, producing customized ads alongside your search results or email messages (in GMail).
- Some decision-making programs can even be used as an agent that acts on your behalf. Basically, the program is configured to make choices based on settings that you supply. It can then respond to a large number of queries for you, or it can take action quickly to respond to time-sensitive queries when you are not available. This is exactly how online auctions work at sites like eBay. You set your maximum bid and an "agent" program automatically submits minimum bids for you until your max is reached. The agent sends you a notice every time it successfully or unsuccessfully bids on your behalf.

# Program Design Worksheet

**Name:**

Input(s):

Output(s):

Example:

Problem Description	(Describe a <u>subset</u> of the problem that will be the focus of the first stage of program development.)
Pseudo-Code:	
	(Implement and Test the above pseudo-code. Use Numbers from the example for testing.)

Repeat: Problem Description (for a subset of the problem), Pseudo code and Implementation.