

EXERCISE SET 1**PROGRAMMING LABORATORY I, ACADEMIC YEAR 2019-2020**

Deadline: 24/11/2019, 23:59

Read before you start

Read the ENTIRE recitation carefully and "plan" your program on paper. If you do not know the meanings of the mean, and the standard deviation, see [here](#) .

Decide what variables you will need, what names you will give them, if they need constants and if so for what quantities, what control structures you will use for each function, how and at what point in the program you will make the required calculations. Each time you complete a step, make sure your program is working properly before proceeding.

If you need clarification or problems, send a message in the e-class discussion area. Warning: posting code in eclass is not allowed.

This work can be done in groups of up to 2 people. You do not need to be in a group with the same person in the lab. You can discuss the exercises with your classmates but you are not allowed to exchange code in any way.

If you work in a team, then before you start working log in to Autolab, select it hw1 and build a team through selection Group Options. Do not leave this process to the last minute!

Start early! Scheduling is always MUCH more time consuming than you expect. Late submissions of exercises are not accepted.

Your exercises will be graded in the following (without special order):

- Correct calculations
- Efficient use of appropriate structures, variables, constants, etc.
- General program formatting (alignment, names of variables and constants, etc.) Compliance with
- specifications
- Effective comments

The use of goto and the use of universal variables is strictly forbidden.

You can assume that input from the keyboard will always be given in the correct format, for example, if a number is requested no string will be given.

Automatic multiple choice test scoring

In this exercise you will write a program that checks the answers of a number of students to a multiple choice test and produces statistics on student performance.

General information

The test contains NUM_QUESTIONS questions. All questions have the same number of options, which correspond to the characters 'A' - 'G'. The first choice is always the 'A' and the rest follow in the same order that appear in the English alphabet. The special character '-' means that a question has not been answered.

The program manages NUM_STUDENTS students taking the test.

For each student, you will need to save the number of correctly answered questions, the number of incorrectly answered questions and their grade. Use a two-dimensional floating-point number table, each line of which will contain the above information about a student.

It is up to you to decide if and what other tables are needed to store the data.

Archives

We give you two files:

- The hw1.h contains the definition of four constants that you will use in your program:
 - NUM_STUDENTS: number of students (default value 50)
 - NUM_QUESTIONS: number of questions (default value 20)
 - MIN_CHOICE: first possible choice (default value 'A')
 - MAX_CHOICE: last possible option (default value 'Z')
- The hw1.c is the file in which you will write your program. Contains the command `#include "hw1.h"` through which its contents hw1.h included in hw1.c so you can use the above constants in your program and command `#include <math.h>` which is necessary for the use of mathematical functions.

You can change the values of the constants defined in hw1.h. No other changes or additional code may be added to this file. You are not allowed to redefine these constants in hw1.c.

Outgoing messages

In all messages ending in a colon, there is a space after it. Where we tell you to print the separator string, print "\n

\n "

Mathematical functions

To compile the math functions you need to add `-lm` (el em hyphen) at the end of the gcc command.

Program operation

Data input

Print line break character and message " Enter number of choices (1-X): " where X the maximum number of options (integer) and read from the keyboard the actual number of options (integer). As long as he is not in the range 1 up to X, printouts and crowd input are repeated. For example, if the MIN_CHOICE is ' A ' and MAX_CHOICE is ' Z ' then the message will be Enter number of choices (1-26):

Based on the value given by the user, calculate the maximum actual selection. For example, if the user gives 5, then the maximum real choice will be ' E '.

Print line break character and message " Enter answer key: "(only once) and read from the keyboard the correct choice for one-by-one question (with a space between them, which is ignored). If input is given that is not in the options range as specified above, then print a change of character and the message " Error. Enter value Y-Z: ", where Y the minimum possible choice and Z the maximum actual selection and repeat the selection reading. The error message is printed and read again until a valid value is given. Continuing the above example, the message should be: Error.

Enter AE:

Print line break character and message " Enter wrong answer penalty: " and read from the keyboard the scoring penalty for incorrect answers, which is a negative floating point number. If the value given is positive, then print a change of character and the message " Error. Enter non-positive value: " and repeat reading the answer. The error message is printed and read again until a valid value is given.

For each student, print a change of character and the message " Enter student choices: "(only once) and read one by one the student's choices (answers) for each question (with a space between them, which is ignored). If you are given an option that is neither in the correct range of options nor '-', then print a change of character and the message " Error. Enter valid choice: " and repeat reading the selection for that question.

Calculations

For each student, calculate the number of right and wrong choices and save these values in the appropriate places on the student data table.

For each student calculate the grade obtained from the formula:

$$\max(0, \frac{(\text{number of correct answers}) + (\text{number of incorrect answers}) * (-\text{penalty})}{(\text{number of questions})} * 10)$$

and save it in the student data table. Calculate the average of the students' grades.

Calculate the standard deviation of

grades resulting from the formula:

$$\frac{\sum_{i=0}^{n-1} (\text{grade}[i] - \text{Average})^2}{n}$$

where n the number of students. To calculate the square root of an expression x and save it to a variable y , you write $y = \text{sqrt}(x)$. To calculate the square of an expression x and store it in a variable y , you write $y = \text{pow}(x, 2)$.

Student grades are always in the range 0-10. Calculate the number of students whose grade belongs to each of the intervals $[0, 1)$, $[1, 2)$, ..., $[8, 9)$, $[9, 10]$ and store these values in an integer table 10, in which cell 0 corresponds to degrees belonging to the interval $[0, 1)$, cell 1 to degrees belonging to the interval $[1, 2)$ and so on. Think of a clever way to fill the table - not with consecutive if statements.

You will find the function useful floor of the Mathematical Library: The Call floor (x) for a real number x returns the largest integer value not greater than x. For example, the floor (3.1) is 3. 0. Similarly, the floor (3) is 3. 0. Because its return type

floor is double, you will need to convert the return value to the corresponding integer with a suitable typecast.

This table will be used later to print the grade histogram.

Function menu

Once the reading and data entry is completed successfully, print function menus repeatedly and manage each function properly.

To be precise, print a line break character in the following menu:

```
[M / m] Print mean
[S / s] Print standard deviation [G / g] Print grade
range
[I / i] Print student info in range [H / h] Draw histogram
[Q / q] Quit
```

and line break character.

Read the user's selection from the keyboard. The correct options are the characters that appear in the menu between brackets. If selected incorrectly, print a change of character and the message " Error. Invalid option. " and repeat menu printing and option reading.

In the selection Q the q, the program terminates.

In the selection M the m, print a dashed character, averages over two decimal places, and a string separator.

In the selection S the s, print the dashed character, the standard two-digit divergence, and the separator string.

In the selection G the g, print line break character and message " Enter grade range: " and read from the keyboard two integers, at least low, high, separated by a hyphen. If the integers read do not belong to the interval $[0, 10]$ or the first is not less than the second, the message is printed and the reading of values is repeated. When valid values are given, calculate the number of students who have a degree in space $[low, high)$ or especially in the event that high is equal to 10, in the interval $[low, high]$. Print a line break character, the number you calculated, and the separator string.

In the selection I the i, print line break character and message " Enter index range: "

and read from the keyboard two integers separated by a hyphen. The integers represent a range of places on the student table. If the first is not less than the second or the range is outside the table limits, the message is printed and the read values are repeated. When valid values are given, print the following information based on the student data in the table positions from the first to the second integer:

- For each student:

- line break character,
- position in the table (index) with a width of 3 characters and zeros at the beginning colon
-
- number of correct answers with a width of 4 characters, without decimal places, comma,
-
- number of incorrect answers with 4 characters width, without decimal places, comma,
-
- number of unanswered questions with 4 characters width, without decimal places, party,
-
- degree with two decimal places and a total width of 6 characters,
- Finally, the separator string.

In the selection H the h, print a histogram showing in a "graphic" way how many students got a grade in each interval [0, 1), [1, 2),..., [8, 9), [9, 10] as calculated earlier. To be precise, for each interval, the program prints:

- left bracket
- the lower limit of the range corresponding to this position with a width of 2 characters, comma,
-
- the upper limit of the width corresponding to this position with a width of 2 characters right
- parenthesis, except in the case of 10, where a right bracket is printed, colon
-
- a gap
- as many stars (*) as the X-groups of students who got points in this range, where
 - X-units = units if the total number of students who took the test is at most 100
 - X-groups = tens if the total number of students who took the test is greater than 100

After printing the histogram, the program prints the separator string.

Instructions for submitting a job in Autolab

1. Add comments at the beginning of the file hw1.c the full names and AEMs of the team members. Please write comments ONLY in English characters.
2. Build a directory by name hw1submit and copy inside this hw1.c
3. Build a text file with a name team.txt and add to it the full names and AEMs of the team members, even if the team consists of one person.
4. Right click on the directory hw1submit and select Compress here as tar.gz. A file with a name will be created hw1submit.tar.gz.
5. Log in to Autolab and select it hw1.
 - If you are a team of two and have not already done so, build a team through it selection Group Options.
6. Do Submit the hw1submit.tar.gz.