

04JCJLZ - COMPUTER SCIENCES - 2015/2016

Laboratory 5

Objectives:

- Write programs able to store and process many values

Technical content:

- Consolidating use of cycles
- Introduction to the use of arrays.

Preferably to be solved in the laboratory:

Exercise 1. Write a program that acquires two positive values from the user and calculates the Greatest Common Divisor (GCD) using the Euclidean algorithm.

Euclidean algorithm or remains method: it proceeds by successive divisions of the larger number by the smaller, replacing at each step the larger value with the smaller and the smaller with the remainder of the integer division. The process ends when the remainder is 0.

Example: A = 34 , B = 18

step 1: $34 \% 18 = 16$

step 2: $18 \% 16 = 2$

step 3: $16 \% 2 = 0 \leftarrow \text{stop!}$

Result: MCD = 2

Hint: First draw the flowchart of the method taking into account that each step corresponds to one iteration and proceed with the implementation of the code.

Exercise 2. Write a program that defines and manipulates an array composed of 10 elements. The program must:

- a. acquire values from the keyboard and store them in the array;
- b. print the content of the array at the end of the acquisition;
- c. calculate and print the average of the values in the array using a variable of type float;
- d. find and print to screen the maximum value and its ordinal position in the array.

Further insight: Consider the case in which the maximum value appears in the array more than once, and print all the relative positions.

- Exercise 3. Write a C program that defines two arrays *v1* and *v2* of *N* elements (all initialized to 0) of type integer and stores the values, acquired from the keyboard, in the arrays, according to the following acceptance rules:
- in *v1* store only positive values, and negative values multiples of 3;
 - in *v2* store only negative values that are odd and not multiples of 3;
 - all other values are ignored;
 - the insertion ends when one of the two arrays is full, at which point the program prints to the screen the content of the arrays acquired.
-

To be solved at home

- Exercise 4. ¹Write a C program that reads from the keyboard at most *N* integer values, with *N* a constant previously defined at will, and save them into an array. The acquisition must proceed until the series of numbers is no longer monotone (i.e., a combination of numbers in ascending or descending order). Print the content of the array at the end of the acquisition.

Example: (*N*=10)

1 4 6 10 4	← The insertion of the value 4 ends the iteration
9 7 6 7	← The insertion of the value 7 ends the iteration
1 2 3 4 5 6 7 8 9 10	← 10 values acquired, the program stops

Hint: First, write a simplified version choosing a single direction of monotony (ascending or descending), then go to the complete solution.

- Exercise 5. Write a C program that scans an array of *N* integers and determines if there is a growing series of three consecutive numbers. In the positive case, the program must print the series of numbers and the position in the array of the first value.
- Exercise 6. Write a C program that analyzes the content of an array containing *N* positive values and searches for duplicates, *N* is a constant defined previously. The program must:
- Acquire the values of the array from keyboard. The acquisition stops when the user enters a negative number, which must not be saved.
 - Start a loop in which it asks for a value, and then save in a second array all the positions in which the value appears in the first array, if present, and then, the program prints the values positions. The loop stops when a negative number is introduced.

Note: a similar technique can be adopted to find the position of the minimum (or maximum) values in an array, in the case when the minimum (or maximum) values can be more than one.

¹ This exercise will be solved using a multimedia format, and its solution will be provided in the course site during the following weeks.