04JCJLZ - COMPUTER SCIENCES - 2015/2016

Laboratory 2

Objectives:

- Define a variable in C language
- Perform simple calculations based on basic arithmetic operators
- Use debugging options for the control of variables during the execution of a program
- Experiment with the use of conditionals statements to make decisions for solving simple math problems.

Technical content

- Assignment of numerical values and debugging using the debugging functions
- Using of arithmetic expressions
- Using preliminary decision constructs like *if* and *if-then-else.*

Preferably to be solved in the laboratory:

Exercise 1. Write a program that defines:

a. one variable of type integer called *int_var* and one variable of type real called *float_var*
b. one variable of type unsigned integer named *uint_var* and one real variable represented with double precision called *double_var*

and furthermore,

c. assign the following values to the variables defined:
    - *int_var* ← -3
    - *float_var* ← 2.5
    - *uint_var* ← *50*
    - *double_var* ←*2/3*

View the contents of the variables by using the debugging option:
Debug→Debbuging windows →Watches

Exercise 2. Write a program that defines three integer variables called *var1*, *var2*, and *tmp*. The program will:

a. Assign the following values to variables defined:
    - *var1*← -3
    - *var2*← 12
b. Using the variable *tmp,* swap the values of *var1* and *var2*.

View the contents of the variables by using the debugging option:
Debug→Debbuging windows →Watches

Exercise 3. Write a program that defines three real variables (*float*) called *length*, *width*, and *perimeter*, corresponding to base, height, and perimeter of a rectangle:

a. Initialize the variables *length* and *width* using the values chosen by the programmer
b. Based on these data, calculate the perimeter of the rectangle and save it in the variable *perimeter*.

View the contents of the variables by using the debugging option:
Debug→Debbuging windows →Watches

To be solved at home

Exercise 4. Write a program that defines two integer variables called *var1*, *var2*. The program will:

a. Assign the following values to variables defined:
   - *var1*← 25
   - *var2*← -53
b. Without using a third variable, exchange the values of the variables *var1* and *var2*.

View the contents of the variables by using the debugging option: Debug→Debbuging windows →Watches

Exercise 5. Write a program that defines three real variables (float) called *price*, *tax*, and *receipt*, and:

a. Assign values chosen by the programmer to *price* and *tax*
b. Based on these data, calculate the price inclusive of taxes (price + price*tax/100) and save it in the variable *receipt.*

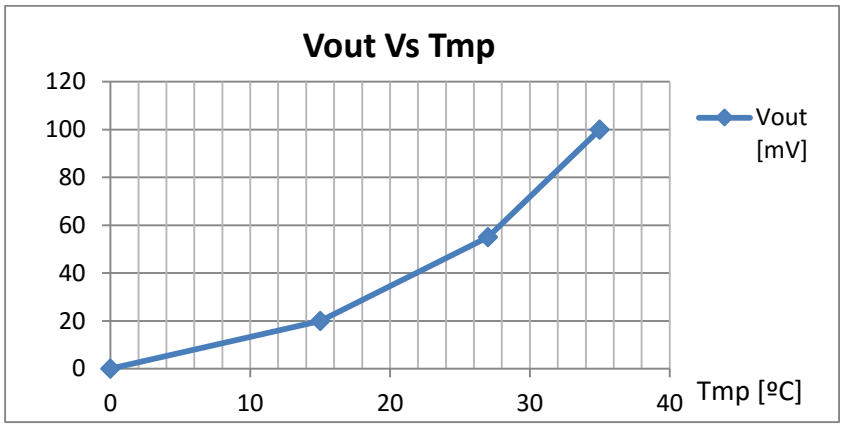
View the contents of the variables by using the debugging option:
Debug→Debbuging windows →Watches

Exercise 6. Based on the flow chart designed to Exercise 2 of Week 1, write a C program for the calculation of the module (absolute value) of a number, in particular the program must:

a. Acquire from keyboard an integer value, positive or negative, and store it in a variable suitably defined
b. Determine, experimenting for the first time the conditional construct *if*, whether that variable contains a negative value and, in this case, transform it in the corresponding positive value
c. Print on screen the final value, which is the module of the acquired value.

Exercise 7. [1]It is required to linearize the output values obtained by a temperature sensor, which provides as output (V_{out}) a voltage signal ranging from 0 to 100 mV, some of the measured values are represented as a polygonal chain:

Tmp [°C]	V_{out} [mV]
0	0
15	20
27	55
35	100



Write a C program which acquires from the keyboard an integer value intended as the output voltage provided by the sensor, expressed in mV, and then the program must calculate and print the temperature value expressed in ℃. The program must use the *if-then-else* construct.

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