VG101 — Introduction to Computer and Programming

Assignment 4

Manuel — UM-JI (Fall 2017)

- MATLAB: write each exercise in a different file
- C/C++: use the provided assignment template
- Include simple comments in the code
- If applicable, split the code over several functions
- Extensively test your code and impove it
- Write a single README file per assignment
- Zip all the files and upload the archive on Canvas

Ex. 1 — *Structure, basic programming*

A complex number is composed of a real part and of an imaginary part. Create a structure to represent complex numbers. Write two functions to calculate the product and the sum of two complex numbers. A third function should ask the user to input two complex numbers together with an operation, and then display the result.

Specifications.

- Provide the user with a sample input
- Demonstrate the running of the program using the sample input

Ex. 2 — *Recursion, basic arithmetic*

Given two integer:

1. Write a recursive algorithm to calculate their GCD.
2. Translate the algorithm into C.

Ex. 3 — *Conditional statements*

1. Write a C function which prompts the user for a functionality and dispatches the work to the corresponding function from question 2 or 3.
2. Write a C function to check whether the letter input by the user is a vowel or a consonant.
3. Write a C function which prompts the user for a number and a character. The character can only be 'b' for bit of 'B' for byte. The function should print all the common data types whose size are equal to the number input by the user. The 'b' and 'B' will determine whether the input number is to be considered as bits or bytes.

Specifications.

- Question 1:
  - Display the list of functionalities to the user
  - Read the user's choice as a digit or a character
- Question 3:
  - At least all the four basic data types should be considered
  - Output exactly one datatype per line

Ex. 4 — *Use of the math library, conditional statements*

Write a C function to find the roots of a quadratic equation whose coefficients are input by the user.

Specifications.

- The polynomial $ax^2 + bx + c$ should be input as the triple "a b c"
- Output exactly one root per line

Ex. 5 — *Low-level computation*

The *Xor swap algorithm* is an algorithm that swaps two values of distinct variables without using any temporary variable.

1. Write a `#define SWAP(a,b)` macro to swap two integers a and b.
2. Write a short C function to demonstrate the previous macro.