Lab 4 : Functions

Exercise 1:

1. Write the C code for a function **`digits_sum`** that reads an integer and returns the sum of its digits.
2. Write the C program that uses **`digits_sum`** to calculate the sum of the digits of a number entered by the user. The sum, in turn, will be converted into the sum of its digits. And so on, until the sum is only one digit long.
   Example : 75 12 3.

Exercise 2:

The Syracuse sequence is defined as follows:

$\mathbf{u_0}$ is a positive integer chosen by the user.
$\mathbf{u_{n+1}} = \mathbf{u_n}$ /2 if $\mathbf{u_n}$ is even.
$\mathbf{u_{n+1}} = 3\ \mathbf{u_n} + 1$ if $\mathbf{u_n}$ is odd.

1. Write a C function (Syracuse) that returns the term u_{n+1} of the Syracuse sequence from the term u_n received as a parameter.
2. Write a C function that uses the Syracuse function to find the smallest value of n for which $u_n = 1$ from a term u_0 received as a parameter.
3. Write a C function that uses the Syracuse function to calculate the largest value reached by the terms u_n from the term u_0 received as a parameter.
4. Write the C program to test these functions.

Exercise 3:

A factorion is a positive integer that is equal to the sum of the factorials of its digits.
Examples:

145 = 1! + 4! + 5!
40585 = 4! + 0! + 5! + 8! + 5!

1. Write a C function factorial() that calculates the factorial of an integer given as a parameter.
2. Write a C function to check if a positive integer given as a parameter is a factorion.
3. Test your functions in a main program for a number entered by the user.

Exercise 4:

1. Write a C function (procedure) to print the prime factors of a number given as a parameter.
2. Test the function in a main program for a number entered by the user.

Example, if the input number is 12, then the output should be “2 2 3”. And if the input number is 315, then the output should be “3 3 5 7”.