

Lab Manual

Course : CSE -103
Credit Title : Structured Programming
Instructor : Dr. Maheen Islam, Associate Professor, CSE Department

Lab-7: C Functions

Exercise 1: A prime number is a natural number greater than 1 that has no positive divisors other than 1 and itself. Write a user define function to check a number whether it is prime or not. If the number is prime print “Prime” else print “Not Prime” and the minimum factor of it. Include your function in a working C program.

OUTPUT

Enter the number: 5

Prime

Enter the number: 9

Not Prime

Minimum factor is 3

Exercise 2: Write a C program to find the factorial of a given number. Write a user defined function for finding the factorial of n.

OUTPUT:

Enter the number to find the factorial

5

The factorial of 5 is 120

Exercise 3: The *Fibonacci numbers* are numbers of an interesting sequence in which each number is equal to the sum of the previous two numbers. In other words,

$$F_i = F_{i-1} + F_{i-2}$$

where F_i refers to the i th Fibonacci number. By definition first two Fibonacci numbers is equal to 1; i.e.,

$$F_1 = 0$$

$$F_2 = 1$$

Hence,

$$F_3 = F_2 + F_1 = 0 + 1 = 1$$

$$F_4 = F_3 + F_2 = 1 + 1 = 2$$

$$F_5 = F_4 + F_3 = 2 + 1 = 3$$

and so on.

Write a C function that will generate first n Fibonacci numbers. Include your function in a working C program.

OUTPUT:

Enter the number: 7

First 7 Fibonacci numbers are: 1, 1, 2, 3, 5, 7 and 12

Exercise 4: Write a C function that calculates the value of weight **z** subject to the following conditions:

$$z = \begin{cases} x^2 - 4y & \text{if } x > y \\ x^2 + 4y & \text{if } x < y \\ (x+y)^{3/4} & \text{if } x = y \end{cases}$$

Then write a C main program that reads the values for **x** and **y** and calls the developed function for calculating the value of weight **z**.

OUTPUT:

Enter x: 3

Enter y: 5

Value of z is: 29

Enter x: 5

Enter y: 2

Value of z is: 17

Exercise 4: Write a C program that reads **n** integer numbers in an array **A** and calls a user defined function that takes two parameters—a pointer to the array **A**, and a second integer indicating the size of the array. The function should be called **sumPositive**. The function should return the sum of all the integers in the array that are greater than 0.

OUTPUT:

Input data size: 5

Input data: -3 10 -5 20 0

Homework:

1. Write a C program to swap the values stored in two different variables. Write a function swap() to swap the elements.
2. Write a function with sample program which will take an array of integer, and the size of the array, and print the elements of array.
3. Write a C program that displays the average of the array values and a table of differences between each array element and the mean. Use a C function to find the average of the array elements.