Introduction to programming
C++ Practical session

ENSIA
Academic Year (2021-2022)

Worksheet 5 « Functions and recursion»

NB. For the following exercises build a minimal program to test functions.

Exercise 1

Write a function "distance" having as parameters 4 doubles xa,ya and xb,yb which represent the coordinates of two points A and B and which returns the distance AB.

Exercise 2

Write a function "**f**" having as parameters a double x and a boolean ok and which returns a double by a return. The function returns with a return the square root of (x-1)*(2-x). The function returns through the variable ok the value true if the function is defined at point x, false otherwise.

Exercise 3

Write a function having an integer as parameter and which returns a boolean via a return: true if the integer is prime and false otherwise.

Exercice 4

Write a function having as parameter an integer n and which returns the n-th prime number: this function will use the function of exercise 3 .

Exercise 5

Write a swap function having 2 integers a and b as parameters and which swaps the contents of a and b.

Exercise 6

Write a function f having as parameters an array t of any size and an integer n indicating the size of the array. f must return a boolean indicating whether there is a value between 0 and 10 in the first n cells of the array.

Exercise 7

Write a function f having as parameters an array t of any size and an integer n indicating the size of the array. f must return a boolean indicating whether there is a value between 0 and 10 in the first n cells of the array t. If f returns true, f must also return the value of the first box in the array between 0 and 10.

Exercise 8

Write a function f having as parameters an array t1 of any size and an integer n indicating the size of the array, as well as an array t2 of the same size as t1. f must return with a return an integer nb indicating the number of values between 0 and 10 in the array t1. f must put in the array t2 the different values between 0 and 10 that it encountered in the array t1.

Exercise 9

Write a function to convert decimal number to binary number .

Exercise 10

The Fibonacci sequence a(0),a(1), a(2), a(3), ..., a(n), ... is defined by
a(0) =1
a(1) = 1
a(n) = a(n-1) + a(n-2), for all n > 2

Write a function "fibonacci(...)" that computes the Fibonacci number corresponding to its positive integer argument.

Exercise 11

Given positive two integers m and n such that m<n, the greatest common divisor of m and n is the same as the greatest common divisor of m and (n-m). Use this fact to write a recursive definition of the function "greatest_common_divisor(...)", which takes two positive integer arguments and returns their greatest common divisor. Test your function in a suitable main program.

Exercise 12

Write a recursive function that calculates the numeric value of a string composed of digits.

Exercise 13

Binary Search is recursively defined algorithm for searching through a sorted array to find a position of a particular value.

The idea is as follows. Suppose we want to find a position of the value v in the previously sorted array a[]. We first look at the value stored half way along a[]. If this happens to be v, we are finished, and we can simply return this middle position. However, if the value at the middle position is less than v, we now know we only have to search the 2nd half of the array, and so we repeat the search procedure just with this half of the list. Similarly, if the value at the middle position is greater than v, we now know we only have to search the 1st half of the array

Write a function with function declaration

int binary_search(int value, int a[], int first, int last);

which searches the values in the array "a[]" from "a[first]" to "a[last]" to find an array element with value "value". If it finds "value", it returns the position of "value" . Otherwise it returns -1.

Exercise 14

The towers of Hanoi is one of the most famous classics problems consisting of three rods and a number of disks of various diameters, which can slide onto any rod . The objective is to move the entire stack to the last rod. The initial stack has 64 disks threaded onto one rod and arranged from bottom to top by decreasing size. We need move the stack from one rod to another under the constraints that exactly one disk is moved at the time and at no time may a larger disk be placed above a smaller disk. Three rods are provided, one being used for temporarily holding disks .Let’s assume that we are attempting to move the disks from rod1 to rod3. Write a program using recursion to move disks and print the precise sequence of rod-to-rod disk transfers.

Exercise 15

Write the following a function :

1-to verify if a number is a power of two
2-to verify if a number is odd or even.
3-to swap two variables without involving a third variable.