Mila University Center
Institute of Mathematics and Computer Science
Department of Computer Science
1st Year LMD - Computer science

SERIES NO. 2 (RECURSION)

Algorithmics and Data Structures 2 - March 2024 -

Exercise 1

Write recursive modules that allow to:

1) Calculate the factorial: N! =1*2*3*4*……..* (N-1) * N
2) Calculate the power: X^N
3) Calculate the sum: $2^2+4^2+6^2+\ldots$ taking N terms

Exercise 2

The Fibonnacci sequence is defined by:

$$\begin{cases} U_0 = U_1 = 1 \\ U_n = U_{n-1} + U_{n-2} \end{cases}$$

Write a recursive function that determines the nth term of the sequence.

Exercise 3

The calculation of the greatest common divisor (GCD) of two positive integers a and b can be done using Euclid's algorithm. This algorithm is based on follwing theorem:

- If a > b and if we carry out the Euclidean division of a by b: a = qb + r with 0 <= r < b then the GCD of a and b is equal to the GCD of b and r.
- To calculate the GCD of a and b, simply iterate this technique: we obtain a zero remainder in a finite number of steps. The last non-zero remainder is then the GCD of a and b.

Write an algorithm to calculate the greatest common divisor of two integers Nbr1 and Nbr2 using recursive function.

Exercise 4

Let V be a vector of integers, write the following recursive modules:

1) The ***ReadVE*** procedure for filling the vector V.
2) The ***Sum*** function that returns the sum of the elements of the vector V.
3) The ***Max*** function that returns the maximum of the vector V.
4) The ***Belong*** function that allows to check if an element exists in the vector or not.

Exercise 5

The dichotomy search for an element in an ordered vector is carried out as follows:

- We divide the array into two approximately equal parts,
- We compare the searched value with the middle element,
- If they are not equal, we only focus on the part containing the desired elements and we neglect the other part.
- We repeat these 3 steps until we have a single element to compare.

Write a recursive function that dichotomy searches for a value ***Val.*** The function returns the rank of this value if it exists and -1 otherwise?