Larbi Ben M'hidi -Oum El Bouaghi- University
Faculty of Exact Sciences and Natural and Life Sciences
Mathematics and Computer Science Department

Academic year: 2023-2024

Level: 1st year "Computer Science & Mathematics"

Module: Algorithmic and Data Structures 2

TD n°3

Pedagogic objective

→ Manipulate recursive sub-algorithms.

Exercise n°1

1st – call (8.5);

before call n=8, x=5
before call n=7, x=5
before call n=6, x=5
before call n=5, x=5
before call n=4, x=5
before call n=3, x=5
before call n=2, x=5
before call n=1, x=5

In the main algorithm
8*5=40

The sub algorithm makes the product of n*x. The instruction **(" after call n=", n, "x= ", x)**; in the product function is never executed.

Exercise n°2

a) Iterative function

```
Function quotient_division ( a ,b :integer):integer;
Begin
Variable S: integer;
S ← 0;
While (a>=b) do
    a←   a- b;
    S←   S+ 1;
Endwhile
  quotient _division  ← S ;
End;
```

b) Recursive function

```
Function quotient _ division_rec ( a ,b :integer):integer;
Begin
If (a<b) then
        quotient_division_rec ←0;
Else
      quotient_division_ rec ← quotient_division_rec (a- b,b )+1;
Endif
End;
```

Exercise n°3

```
Algorithm Calculation ;
Variables X,Y, P : integer ;
Function GCD (a, b : integer) : integer ;
Begin
If (a = b)  then // Particular case = Stopping criterion.
       GCD ← a
Else
  If (a > b) then  // General case
            GCD ← GCD (a-b, b);
  else   // General case
            GCD ← GCD (a, b-a);
```

```
    Endif
  Endif
End;
Begin
Read (X ,Y);
If (Y≤ 0) Or (X ≤ 0) then
   Write (“Numbers are not strictly positive”);
Else
    P ← GCD (X, Y) ;
    Write (“THE PGCD of”, X, Y, ‘ =', P) ;
Endif
END
```

Exercise n°4

Type Tab = array [1..50] integer ;

a. Recursive function to calculate the sum of the elements of an array

```
Function Sum_tab ( var T: Tab, n: integer ) : integer ;
/* n is the number of elements in the array
Begin
  If (n=1) then
          Sum_tab ← T[1];
   else
          Sum_tab ← T[n] + Sum_tab (T, n-1);
    Endif
End;
```

b. Recursive procedure to reverse the elements of an array

```
Procedure inverse_ tab (var T:tab; d, f: integer);
Variable x: integer;
Begin
If (d<f) then
```

```
    x ← t[d ];
    t[d ]← t[f];
     t[f ]←x;
    inverse_tab (T,d +1,f-1);  // Recursive call
Endif
End;
```

Note

- **d:** start of the table to be reversed.
- **f:** end of the table to be reversed.