La récursivité

Ex. 1: Déterminez l'affichage des programmes suivants:

```
program Enigma1;
var x: integer;
 Procedure Enigma(Var x:integer;
                  y : integer);
   begin
     if y > 0 Then
     begin
       x := x+1;
       y := y-1;
       Writeln ('Avant l''appel X= ',x,'
                     Y= ',y);
       Enigma(x,y);
       Writeln ('Apres l''appel X= ',x,'
              Y= ',y);
     end;
   end;
begin
  x := 1;
  Enigma(x,0);
  x:= 0;
  Enigma(x,1);
  x := 5;
  Enigma(x,3);
end.
```

Ex. 2: Fonction de Fibbonacci

```
program Fibonaci;
  var i : integer;
 function Fib_rec(m:integer):integer;
   begin
     if m < 3 Then
       Fib_rec :=1
       else
    Fib_rec := Fib_rec(m-1)+ Fib_rec(m-2);
   end;
 function Fib_it(m:integer):integer;
   var i,prev1,prev2,t:integer;
   begin
   prev1 :=1; prev2 :=1;
   t :=1;
     for i := 3 to m do
     begin
       t := prev1 + prev2;
       prev2 := prev1;
       prev1 := t;
     end;
    Fib_it := t;
   end;
begin
for i := 1 to 12 do
  writeln ( i:2, Fib_rec(i):5,
                 Fib_it(i):5);
end.
```

Ex.3 Tri par QuickSort

```
Program QuickSortTableau;
  Const   Nmax = 20;
 Type
  Tableau = Array[1..Nmax] of Real;
 Var
  A : Tableau;
  N: integer;
Procedure LireA(var N:integer;
          Var A:Tableau);
   var i : integer;
  Begin {LireA}
  Repeat
Write ('Entrez le nombre des elements: ');
    Readln(N);
  Until (N > 0) And (N <= Nmax);
Writeln ('Entrez ', N,
          ' nombres reeles:');
  For i := 1 To N Do Read(A[i]);
  Readln;
  End;{LireA}
Procedure QuickSort(var A:Tableau;
N:Integer);
procedure qs(idebut,ifin:integer);
var i,j :integer; t, pivot:real;
begin {qs}
  i:=idebut; j := ifin;
   pivot := A[(i+j) div 2] ;
   repeat
     while A[i] < pivot do i := I+1;
     while pivot < A[j] do j := j-1;
     if i <= j then
       begin
         if i < j then
           begin
             t := A[i]; A[i] := A[j];
                        A[j]:= t;
           end;
           i := i+1; j:= j-1;
       end;
     until i>j;
     if idebut < j then qs(idebut,j);
     if i < ifin then qs (i,ifin);
  end; {qs}
Begin {QuickSort}
  qs(1,N);
End;{QuickSort}
  Procedure AffichA(var A: Tableau; N:
integer);
  var i : integer;
  Begin {AffichA}
   For i := 1 To N Do Write(A[i]:8:2);
   Writeln;
  End; {AffichA}
 Begin {programme}
```

```
  LireA(N,A);{Lire les donnees}
  AffichA(A,N);
  QuickSort(A,N);{Tri}
   {affichage}
   Write ('Apres le tri');
  AffichA(A,N);
  readln;
  End.{programme}
```

Ex.4. Les 8 reignes

```
program HuitReignes;
   var i : integer; succes: boolean ;
   ligne: array [1..8] of integer;
   col : array [1..8] of boolean;
   d1 : array [-7..7] of boolean;
   d2 : array [2..16] of boolean;
   procedure essay(i:integer; var succes :
  boolean);
   var j : integer;
   begin
    j:=0;
    repeat
      j:= j+1; succes := false;
      if col[j] and d1[i-j]
                and d2[i+j] then
      begin {registration}
        ligne[i] := j;
        col[j] := false; d1[i-j] := false;
        d2[i+j] :=false;
        if i < 8 then
          begin
            essay(i+1,succes);
            if not succes then
            begin {liberer}
              col[j] := true;
              d1[i-j] := true;
              d2[i+j]:= true;
            end;
          end else succes := true;
        end;
     until succes or (j=8);
  end; {essay}
  begin
    for i := 1 to 8 do col[i] := true;
    for i := -7 to 7 do d1[i] := true;
    for i := 2 to 16 do d2[i] := true;
    essay(1,succes);
    if succes then
       for i := 1 to 8 do
   write(ligne[i]:4);
    writeln;
    readln;
 end.
```