

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

```

Program p;
var i,j,k : integer;
Function f1(x,y: integer) : integer;
var i: integer;
Begin
i := x+1;
f1 := x+y+i;
Writeln('F1: ',i,j,k,x,y);
End;
Procedure p1(var y: integer; x :
integer);
var j,k : integer;
Begin
y := x-2;
j := f1(x,x-1) + f1(x,y)
x := x+1; k := j-1;
y := y+x-j+k;
writeln('P1: ',i,j,k,x,y);
End;
Begin {programme}
i := 10; j := 5; k:= 3;
p1(i,j);
Writeln('P: ',i,j,k);
End.

```

**Ex. 2: Fonction qui calcule le nombre supérieur**

```

Program Fonctions;
Var x,y : Integer;
Function Max(a, b : Integer) : Integer;
Begin
if a >= b Then
Max := a
else
Max := b;
End;
Begin { programme }
x := Max(2,5);
y := Max(7,3);
WriteLn(Max(x,y));
End.

```

**Ex.3 Tri par sélection**

```

Program Tri_selection;
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 SortA(var A:Tableau; N:Integer);
var k,im : integer; m:Real;
Begin {SortA}
For k := N Downto 2 Do
Begin
MaxA(A,k,m,im);
if im <> k Then
Begin
A[im] := A[k]; A[k] := m;
End;
End;
End;{SortA}
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}
SortA(A,N);{Tri}
{affichage}
Write ('Apres le tri');
AffichA(A,N);
End.{programme}

```

**Ex.4. Fonction qui trouve le ppcm**

```

Function ppcm(a,b:integer):integer;
Var
ma,mb : Integer;
Begin
ma := a; mb := b;
While ma <> mb Do
If ma < mb Then
ma := ma + a
Else
mb := mb + b;
ppcm := ma;
End;

```

**Ex.5 Ensemble des fonctions et procédures qui réalisent l'arithmétique des nombres rationnels.**

```

Type rationnel = record
nom,denom : integer
end;
function ppcm(m,n:integer):integer;

```

```

function pgcd(m,n:integer):integer;
procedure reduire(a,b:integer, var
ka, kb, den :integer);
Var
ma, mb : Integer;
Begin
ma := a; mb := b; ka := 1; kb:=1;
While ma <> mb Do
If ma < mb Then begin
ma := ma + a; ka := km+1; end
Else begin
mb := mb + b; kb := kn+1 ;
end ;
den := ma;
end ;
procedure reduction (var a : rationnel) ;
var
den : integer ;
begin
den := pgcd(a.nom,a.denom);
if den > 1 then
begin
a.nom := a.nom div den;
a.denom := a.denom div den;
end;
end;
procedure plus (a,b:rationnel; var
r:rationnel);
var
ka, kb, den:integer;
begin
reduire(a.denom,b.denom,ka, kb, den);
r.nom := ka*a.nom + kb*b.nom;
r.denom := den;
end;

```

**Ex .5 Une procédure combinée qui trouve la valeur extrême selon un ordre défini par une fonction.**

```

{$F+}
Program Extremales;
const
Nmax = 20;
Type
Tableau = Array[1..Nmax] of Real;
Ordre = Function (x,y : real) : boolean;
Var
A : Tableau;
N,imax: integer;
max : real;
Function Croissant (a,b : real):boolean;
Begin
Croissant := a <= b;
End;
Function Decroissant (a,b :
real):boolean;
Begin
Decroissant := a >= b;
End;

```

```

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 ExtremeA(var A:Tableau;
N:Integer;var max:real; var
imax:integer;
Compare : Ordre);
var i : integer;
Begin {ExtremeA}
max := A[1]; imax := 1;
for i := 2 To N Do
If not Compare (A[i],max) Then
Begin
max := A[i]; imax := i;
End;
End; {ExtremeA}
Begin {programme}
LireA(N,A);{Lire les donnees}
ExtremeA(A,N,max,imax,Croissant); }
Writeln ('La valeur maximale = ', max:6:2,
' et sa place est ',imax);
ExtremeA(A,N,max,imax,Decroissant);
Writeln ('La valeur minimale = ', max:6:2,
' et sa place est ',imax);
End.{programme}

```

**Ex. 6 Programme qui lit les numeros, couleurs et le poids d'un certain nombre de balles et trouve le couleur de la plus lourde balle**

```

program Jeu_de_balles;
const ESP = '';
maxballes = 20;
type
balle = record
numero : integer;
couleur : string[20];
poids : real;
end;
balles = array [1..maxballes] of balle;
var
sac : balles;
pos,N : integer;
procedure LireBalle(var b:balle);
begin
with b do
begin
readln(numero, poids, couleur);

```

```

end
end;
procedure LireBalles(var A:balles; var N:integer);
  var i : integer;
begin
  repeat
    Writeln('Tapez le nombre des balles: ');
    readln(N);
  until (N>0) and (N<=maxballes);
  for i := 1 to N do
  begin
    write ('Tapez le numero, le poids et le couleur de la ',i,'-
eme balle: ');
    LireBalle(A[i]);
  end;
end;
procedure AfficheBalles(A:balles; N:integer);
var i:integer;
begin
  writeln ('position':8,'numero':8, 'couleur':10,'poids':8);
  for i:=1 to N do with A[i]do
    writeln(i:8,numero:8,couleur:10,poids:8:2);
  end;
end;
function PosLourde(var A:balles;N: integer):integer;
  var i,pos:integer;
  maxpoids :real;
begin
  pos:=1; maxpoids:=A[1].poids;
  for i:=2 to N do
    with A[i] do
      if maxpoids < poids then
        begin
          maxpoids := poids; pos :=i;
        end;
  PosLourde := pos;
end;
begin
  LireBalles(sac,N);
  AfficheBalles(sac,N);
  pos := PosLourde(sac,N);
  writeln('La couleur de la plus lourde balle est : ',
sac[pos].couleur);
end.

```