



---

# TurboPascal - Prépa HEC Ipecom

## Exemples - Exercices

vendredi 29 mars

---

## 1 Exemples

### 1.1 Procédure simple

```
procedure pause;
begin
  writeln('Appuyez sur RETURN pour continuer');
  readln;
end;
```

### 1.2 Variables locales

```
procedure tirage;
var n1,n2 : integer;
    double : boolean;
begin
  n1:=random(6)+1; n2:=random(6)+1; double:=(n1=n2);
  writeln('Le tirage donne ',n1,' et ',n2);
  if n1=n2 then writeln('Double ',n1,'!!! Jour de chance...');
end;
```

### 1.3 Variables globales

```
program bidon;
const n : 10;
var a,b,c : integer;
procedure affiche;
begin
  writeln('a : ',a,' b : ',b,' c : ',c);
end;
```

```
begin
  for a:=1 to n do
    for b:=1 to a do
      for c:=1 to b do
        affiche;
      end;
    end;
  end.
```

### 1.4 Argument passés par valeur

```
procedure salut(nom : string, numero : integer);
begin
  writeln('Bonjour ',nom,'. Vous etes le numero ',numero);
end;
```

## 1.5 Arguments passés par adresse

```

procedure tri_croissant_int(var a,b : integer);
  var c : integer;
  begin
    if a>b then begin c:=a; a:=b; b:=c; end;
  end;

```

---

```

program test;
var n1,n2,m,s : integer;
procedure plus_grand_somme_int(a,b : integer; var c,d : integer);
  begin
    if a>b then c:=a; else c:=b; end;
    d:=a+b;
  end;

begin
  writeln('2 nombres entiers SVP'); readln(n1,n2);
  plus_grand_int(a,b,m,s);
  writeln('Le plus grand est ',m,' et la somme vaut ',s);
end.

```

## 1.6 Fonction

```

function f(a,b : real) : real;
  begin
    f:=log(sqr(a+b)/(sqr(a)+sqr(b)));
  end;

```

---

```

function fact(n : integer) : integer;
  var i : integer;
  begin
    fact:=1;
    for i:=2 to n do fact:=fact*i;
  end;

```

## 1.7 Fonction, procédure et imbrication

```

prog exemple;
const n=10;
type ligne=array[1..n] of integer;
var a : ligne;
    i,s1,s2,s3 : integer;

function puiss_int(a,b : integer) : integer;
  var i : integer;
  begin
    puiss_int:=1;
    for i:=1 to b do puiss_int:=puiss_int*a; end;
  end;

```

```
procedure somme(t : ligne; expo : integer; var som : integer);
  var i : integer;
  begin
    som:=0;
    for i:=1 to n do som:=som+puiss_int(t[i],expo);
  end;

begin
  for i:=1 to n do a[i]:=i;
  somme(a,1,s1); somme(a,2,s2); somme(a,3,s3);
  writeln('somme des ',n,' premiers entiers : ',s1);
  writeln('au carre : ',s2);
  writeln('au cube : ',s3);
end.
```

## 2 Exercice de concours

### 2.1 HEC 99 Maths1 opt. sci.

$R$  est une matrice définie dans une question précédente, par la propriété  $W_{n+1} = RW_n$ , où  $W_n$  est un vecteur à  $N$  éléments. On a  $R = [r_{i,j}]$ , où  $r_{i,i} = 1 - \beta\Delta i(N - i)$ ,  $r_{i,i+1} = \beta\Delta(i + 1)(N - i - 1)$  et  $r_{i,j} = 0$  si  $j \notin \{i, i + 1\}$  [ $\beta$  et  $\Delta$  sont des données] [ $r_{N,N+1}$  n'existe pas, bien sur].

Dans le préambule d'un programme écrit en TurboPascal on a défini:

```
Const beta='constante fixée par l'utilisateur';
      N='constante fixée par l'utilisateur';
      Delta='constante fixée par l'utilisateur';
Type vecteur=array[1..N] of real;
```

- Ecrire le corps de la procédure `Procedure Calcul1(Var V : vecteur);` qui doit retourner dans  $V$  le résultat de  $RV$ .
- Ecrire le corps de la procédure `Procedure Calcul2(Var V : vecteur, i : integer);` qui doit retourner dans  $V$  le contenu de  $W_i$  défini plus haut [ $W_0$  est entièrement nul, sauf son dernier élément qui vaut 1]. Cette procédure peut utiliser la procédure précédente.

### 2.2 HEC 99 Maths2 opt. sci.

On suppose avoir défini dans un programme Pascal: `Type Tableau=Array[1..100] of integer;`

- Ecrire une procédure `procedure Tirage(var C:Tableau);` permettant de simuler le tirage avec remise de 100 boules dans une urne contenant des boules de couleur  $C_1$  ou  $C_2$  ou  $C_3$  en proportion respectivement  $1/4$ ,  $1/4$ ,  $1/2$ . L'élément  $C[i]$  vaut 1, 2 ou 3 et représente la couleur de la  $i$ ème boule tirée. On utilisera la fonction `random : random(4)` retourne un entier aléatoire compris entre 0 et 3.
- Ecrire une fonction `Difference` de paramètre  $C$  qui retourne la valeur de  $X_1 - X_2$  [on a défini précédemment dans le sujet  $X_i$  comme le nombre de boules de couleur  $C_i$  à l'issue des tirages].

### 2.3 HEC 99 Maths3 opt. éco.

Dans le programme TurboPascal suivant, la fonction `RANDOM` renvoie, pour un argument  $M$  de type `INTEGER`, un nombre entier aléatoire de l'intervalle  $[0; M - 1]$ .

```
PROGRAM simulation;
VAR T : ARRAY[1..20001] OF INTEGER;
    U,S,i,n : INTEGER;
    coincide : BOOLEAN;
PROCEDURE X;
BEGIN
  RANDOMIZE; {initialisation de la fonction RANDOM}
  FOR i:=1 TO 20001 DO T[i]:=1+RANDOM(20000);
```

```

END;

BEGIN
X; i:=1; coincide:=FALSE;
REPEAT
  i:=i+1; S:=0;
  WHILE (S<i-1) AND NOT coincide DO
    BEGIN
      S:=S+1;
      IF T[S]=T[i] THEN coincide=TRUE;
    END;
  UNTIL coincide=TRUE;
U:=1;
FOR n:=1 TO i DO write(T[n],', ',');
WRITELN; WRITELN('U = ',U); WRITELN('S = ',S);
END.

```

- Que fait la procédure X?
- Que représentent les variables U et S à la fin du programme?
- Pourquoi est-il certain que le nombre de passages dans la boucle REPEAT ... UNTIL est fini?

## 2.4 ESC01 Maths2 opt. éco.

On considère le turboPascal suivant, où RANDOM(10) désigne un nombre entier tiré au hasard par l'ordinateur dans l'intervalle [0; 9] (la procédure RANDOMIZE sert à initialiser la fonction RANDOM):

```

PROGRAM Tirage;
VAR a,b,c : INTEGER;

BEGIN
  RANDOMIZE;
  a:=RANDOM(10)+1; b:=RANDOM(1)+1;
  IF a>b THEN
    BEGIN
      c:=a; a:=b; b:=c;
    END;
  IF a<b WRITELN('(',a,',',b,')');
END.

```

- Que fait l'ordinateur dans le cas où les variables a et b contiennent toutes deux le même nombre?
- Qu'affiche l'ordinateur dans le cas où les variables a et b contiennent respectivement les nombres 3 et 5?
- Qu'affiche l'ordinateur dans le cas où les variables a et b contiennent respectivement les nombres 10 et 1?

## 2.5 ESC01 Maths3 opt. éco.

Dans le langage informatique Pascal, la fonction `random` renvoie, pour un argument  $m$  entier vérifiant  $m \geq 1$ , un nombre entier aléatoire compris entre 0 et  $m - 1$  (cette fonction est initialisée au début du corpa par la procédure `randomize`). On rappelle que, dans l'exécution d'une boucle `for i:=n downto 2, i` prends successivement les valeurs  $n, n - 1, \dots, 2$ .

Dans un programme écrit en Pascal figurent la déclaration `type tableau=array[1..5] of integer;` et la procédure

```
procedure aleatoire(var A:tableau);
var aux,i,alea : integer;
begin
  for i:=1 to 5 do A[i]:=i;
  for i:=5 downto 2 do
    begin
      alea:=random(i)+1;
      aux:=A[alea]; A[alea]:=A[i]; A[i]:=aux;
    end;
  end;
```

- – On suppose que les valeurs successives de `alea` sont 4, 2, 3 et 2. Donner les valeurs de `A[1]`, `A[2]`, `A[3]`, `A[4]` et `A[5]` à la fin de l'exécution de la procédure.
- Quelles valeurs successives doit prendre `alea` pour obtenir, à la fin de l'exécution de la procédure, le tableau `A[1]=3`, `A[2]=5`, `A[3]=2`, `A[4]=4` et `A[5]=1`.
- Expliquer pourquoi cette procédure permet de simuler un tirage sans remise de 5 jetons, aussi appelée permutation de l'ensemble  $\{1, \dots, 5\}$ .
- Ecrire une fonction d'en-tête `function T(A:tableau):integer;` qui renvoie le nombre de sous-suites croissantes du tableau `A` correspondant à une permutation de  $\{1, \dots, 5\}$ .
- On suppose que le programme contient les déclarations `var A:tableau; var k:integer; var S:real;` et que le corps du programme est le suivant:

```
begin
  randomize;
  S:=0;
  for k:=1 to 1000 do
    begin
      aleatoire(A);
      S:=S+T(A);
    end;
  S:=S/1000;
  writeln(S);
end.
```

Après exécution du programme la valeur affichée de  $S$  est 2.98. Ce résultat est-il étonnant?