

partie III : Les tableaux a 2 dimensions
(Les matrices)
(تبادلات)

III-1 - Definition :

un tableau bidimensionnel est constitué de lignes et de colonnes, Les quantités étant alors définies dans ce cas par :

lignes : de début ... fin

colonnes : début ... fin

Exemple :

soit A une matrice de 2 lignes et 3 colonnes

	col1	col2	col3
ligne 01	1	7	0
ligne 02	5	-5	10

les lignes sont déterminées par les indicateurs j et les colonnes par les indicateurs i

une valeur de la matrice est donnée par :

$$A[2, 3] = 10.$$

\downarrow
 2 = ligne
 \downarrow
 3 = colonne

$$\text{ou : } A[1, 2] = 7$$

\downarrow
 1 = ligne
 \downarrow
 2 = colonne

III-2: déclaration d'une matrice au sein d'un programme

*déclaration 1:

Soit A une matrice de N lignes et M colonnes avec
 $N = 50$ et $M = 100$
C++

A: Array [1..50, 1..100] of integer;

*déclaration 2:

C++

A: Array [1..N, 1..M] of integer;

const

$n = 50;$

$m = 100;$

III-3: Lecture et affichage d'une matrice:

III-3-1: La lecture des éléments d'une matrice

```
for i := 1 to n do  
  Begin  
    for j := 1 to m do  
      Begin  
        write ln ('M [i,j] = ');  
        readln (M[i,j]);  
      end;  
    end;  
  end;
```

} lecture des
éléments
d'une matrice
 $M(n, m)$

III-3-2: L'écriture (Affichage) des éléments de la matrice:

```

for i := 1 to n do
  for j := 1 to m do
    write ln ( 'Les éléments de la matrice sont' M[i,j] )
  end;

```

} Affichage
} $M(n,m)$

III-3-3: Exemple d'application:

* A partir de la matrice $M_1(n, m)$ saisie à partir du clavier, construire la matrice M_2 , tel que: $M_2(i, j) = (M_1(i, j))^2$ puis afficher les éléments de la matrice M_2

* Solution:

program Exemple;

var

M_1, M_2 : Array [1..50, 1..70] of integer; } déclaration
 i, j, N, m : integer;

Begin

write ln ('entrez la taille de la matrice'); } introduction
 de la taille

read ln (N, m);

for i := 1 to N do

Begin

for j := 1 to m do

Begin

} introduction
 des
 éléments

```
writeLn ('M1[i,j] = ');  
ReadLn (M1[i,j]);
```

```
end;
```

```
end;
```

```
for i := 1 to N
```

```
begin
```

```
for j := 1 to m
```

```
begin
```

```
M2[i,j] := SQRT (M1[i,j]);
```

```
end;
```

```
end;
```

```
for i := 1 to N do
```

```
begin
```

```
for j := 1 to m do
```

```
begin
```

```
writeLn ('les elements de M2 sont', M2[i,j])
```

```
end;
```

```
end;
```

```
Readkey;
```

```
End.
```

construction
de la matrice
 M_2 .

Affichage
de la matrice
 M_2 dans

la Boucle

III - 4 : des Matrice particulières :

III - 4 - 1 : Matrice carrée

La matrice carrée définit par une matrice dont le nombre de ligne = le nombre de colonne ($n = m$), caractérisée par la diagonale (جسلی)

*ex eple : $a = \begin{bmatrix} 4 & 7 & 5 \\ 8 & -3 & 4 \\ 9 & 2 & 1 \end{bmatrix}$ ← matrice carrée

III - 4 - 2 : Matrice triangulaire supérieure

c'est une matrice carrée dont les éléments ayant ($\forall i > j$) sont nul (égal 0) $\forall i > j \Rightarrow a_{[i,j]} = 0$

*ex eple :

$a_{sup} = \begin{bmatrix} 4 & 7 & 5 \\ 0 & 3 & 4 \\ 0 & 0 & 1 \end{bmatrix}$ ← matrice triangulaire supérieure

III - 4 - 3 : Matrice triangulaire inférieure

c'est une matrice carrée dont les éléments ayant cette condition ($i < j$) sont nul ; $\forall i < j \Rightarrow a_{[i,j]} = 0$

ex eple :

$a_{inf} = \begin{bmatrix} 4 & 0 & 0 \\ 8 & -3 & 0 \\ 9 & 2 & 1 \end{bmatrix}$ ← matrice triangulaire inférieure

III-5- Exercice d'Application et execution

* les notions de l'exercice 04 de la serie de TD

III-5-1 solution:

```
program Exo4;
```

```
uses crt;
```

```
var
```

```
A: Array [1..50, 1..70] of integer;
```

```
i, j, N, M, Nbp, Nbn, Sp, Sn: integer;
```

de'claration

Variables

```
Begin
```

```
writeln (entrez la taille de la matrice);
```

```
readln(N, M)
```

saisie
taille de la
matrice

```
for i := 1 to N do
```

```
for j := 1 to M do
```

```
Begin
```

```
writeln('A[i, j] =');
```

```
readln(A[i, j])
```

saisie
les elements
de la
matrice

```
end;
```

```
for j := 1 to M do
```

```
Begin
```

```
Nbp := 0, Sp := 0;
```

```
for i := 1 to N do
```

Begin

if $A[i, j] > 0$ then

Begin

$Nbp := Nbn + 1;$

$Sp := Sp + A[i, j];$

end;

end;

writeln ('le nombre des elements positifs', Nbp);

writeln ('la ^{somme} ~~nombre~~ des elements ~~negatif~~ positifs', Sp);

end;

for $i := 1$ to N do

Begin

$Nbn := 0, Sn := 0;$

for $j := 1$ to m do

Begin

if $A[i, j] < 0$ then

Begin

$Nbn := Nbn + 1;$

$Sn := Sn + A[i, j];$

end;

end;

writeln ('le nombre des elements positifs', Nbn);

writeln ('le nombre des elements Negatif', Sn);

end;

ReadKey;

End.

- III - 5 - 2 : Execution détaillée

Les éléments de la matrice sont par exemple les

$$n = 3$$

$$m = 3$$

$$\begin{bmatrix} 7 & 15 & -3 \\ 4 & 1 & 10 \\ 0 & -8 & -5 \end{bmatrix}$$

Nbp : nombre des éléments positifs par colonne

Sp : somme des éléments positifs par colonne

Nbn : nombre des éléments négatifs par ligne

~~Sn~~ : ~~nombre~~ somme des éléments négatifs par ligne

ds ce cas Apres execution du programme, on

obtient les résultats suivants

$$Nbp = 3$$

$$Nbp = 2$$

$$Nbp = 1$$

$$Sp = 11$$

$$Sp = 16$$

$$Sp = 10$$

$$Nbn = 1$$

$$Nbn = 0$$

$$Nbn = 2$$

$$Sn = -3$$

$$Sn = 0$$

$$Sn = -13$$