

Corrigé d'Examen Final : Sujet -A-

Exercice 1 :

Soit TAB un tableau de N ($N \leq 100$) entiers.

Ecrire un algorithme qui permet de :

- Remplir le tableau TAB.
- Vérifier et afficher si le tableau est trié ou non dans l'ordre **croissant**.
- Si le tableau **est trié** :
 - Déterminer et afficher le plus grand nombre **pair** s'il existe.
 - La moyenne des nombres **positifs** s'ils existent.

Remarques :

- Le tableau TAB peut ne contenir aucun nombre pair. Dans ce cas, le message à afficher est : "Aucun nombre pair existe dans le tableau TAB".
- Le tableau TAB peut ne contenir aucun nombre positif. Dans ce cas, le message à afficher est : "Aucun nombre positif existe dans le tableau TAB".

Corrigé Exercice 1 :

09 points

Algorithme exo_01 ;

var TAB : tableau [1..100] entier ;

N, i, Somme, Nbre_pos, Max_pair : entier ;

Début

répéter

lire(N) ;

Jusqu'à ((N > 0) et (N ≤ 100)) ;

Pour i de 1 à N Faire

Ecrire ("TAB[" , i , "] = ") ;

Lire(TAB[i]) ;

FPour

/* vérification si le TAB est trié ou non */

i ← 1 ;

Tant Que ((i < N) et (TAB[i] ≤ TAB[i+1])) Faire

i ← i+1 ;

FTQue

Si (i = N) Alors

Ecrire("Le Tableau TAB est trié") ;

/* Recherche le plus grand nombre pair s'il existe */

i ← 1 ;

Tant Que ((i ≤ N) et (TAB[i] mod 2 = 1)) Faire

i ← i+1 ;

FTQue

Si (i = N+1) Alors

Ecrire("Aucun nombre pair existe dans le tableau TAB") ;

Sinon

Max_pair ← TAB[i] ;

i ← i+1 ;

Tant Que (i ≤ N) Faire

Si ((TAB[i] mod 2 = 0) et (TAB[i] > Max_pair)) Alors

Max_pair ← TAB[i] ;

Fsi

i ← i+1 ;

FTQue

Ecrire("Le plus grand nombre pair est : " , Max_pair) ;

FSi

Corrigé d'Examen Final : Sujet -A-

/ Calcul de la moyenne des nombres positifs s'ils existent */*

```

i ← 1 ;
Tant Que ((i ≤ N) et (TAB[i] < 0)) Faire 0,50
    i ← i + 1 ;
FTQue

Si (i = N + 1) Alors
    Ecrire ("Aucun nombre positif existe dans le tableau TAB") ; 0,50
Sinon
    Somme ← TAB[i] ;
    Nbre_pos ← 1 ;
    i ← i + 1 ;

    Tant Que (i ≤ N) Faire 0,25
        Si (TAB[i] ≥ 0) Alors 0,25
            Somme ← Somme + TAB[i] ; 0,25
            Nbre_pos ← Nbre_pos + 1 ; 0,25
        Fsi
        i ← i + 1 ;
    FTQue
    Ecrire ("La moyenne des nombres positifs est : ", Somme/Nbre_pos) ; 0,50
Fsi
Sinon
    Ecrire ("Le Tableau TAB n'est pas trié") ; 0,50
Fsi
Fin.
    
```

Exercice 2 :

Soit MAT une matrice N*M (N ≤ 15 et M ≤ 20) d'entiers.

Ecrire un algorithme qui permet de :

- Remplir la matrice MAT par des valeurs strictement **positives**.
- Déterminer l'intervalle des valeurs de la matrice MAT (sous forme : [borne_inf...borne_sup]).
- Déterminer et afficher le **nombre de valeurs supérieurs** à la valeur de la moyenne de la matrice.

Corrigé Exercice 2 :

```

Algorithme exo_02;
var MAT: tableau [1..15, 1..20] entier ;
    N, i, j, Somme, Nbre_pos, Max_pair : entier ;
    
```

Début

/ Remplir la matrice MAT par des valeurs strictement positives. */*

Répéter

```

    Ecrire ("Donner le nombre de lignes N = ") ;
    lire (N) ;
    Ecrire ("Donner le nombre de colonnes M = ") ;
    lire (M) ;
    Jusqu'à ((N > 0) et (N ≤ 15) et (M > 0) et (M ≤ 20)) ;

    Pour i de 1 à N Faire
        Pour j de 1 à M Faire
            Répéter
                Ecrire ("MAT[" , i , ", " , j , "] = ") ;
                lire (MAT[i, j]) ;
                Jusqu'à (MAT[i, j] > 0) ;
            FPour
        FPour
    
```

11 points

Corrigé d'Examen Final : Sujet -A-

/ Déterminer l'intervalle des valeurs de la matrice MAT (sous forme : [borne_inf...borne_sup]). */*

```
borne_inf ← MAT[1,1] ; 0,25
borne_sup ← MAT[1,1] ; 0,25
Pour i de 1 à N Faire 0,25
  Pour j de 1 à M Faire 0,25
    Si (borne_inf > MAT[i,j]) Alors ) 0,50
      borne_inf ← MAT[i,j] ;
    Fsi
    Si (borne_sup < MAT[i,j]) Alors ) 0,50
      borne_sup ← MAT[i,j] ;
    Fsi
  FPour
```

FPour
Écrire ("L'intervalle des valeurs de la matrice MAT est : [", borne_inf, "..", borne_sup, "]"); 0,75

/ Déterminer et afficher le nombre de valeurs supérieurs à la valeur de la moyenne de la matrice */*

```
Somme ← 0 ; 0,50
Pour i de 1 à N Faire 0,25
  Pour j de 1 à M Faire 0,25
    Somme ← Somme + MAT[i,j] ; 0,50
  FPour
FPour
Moy ← Somme / (N*M) ; 0,50
Nbre_Sup_Moy ← 0 ; 0,50
Pour i de 1 à N Faire 0,25
  Pour j de 1 à M Faire 0,25
    Si (MAT[i,j] > Moy) Alors 0,50
      Nbre_Sup_Moy ← Nbre_Sup_Moy + 1 ; 0,50
    Fsi
  FPour
```

FPour
Écrire ("Le nombre de valeurs supérieurs à la valeur de la moyenne de la matrice est : ", Nbre_Sup_Moy); 0,75

Fin.

Corrigé d'Examen Final : Sujet -B-

Exercice 1 :

Soit TAB un tableau de N ($N \leq 150$) entiers.

Ecrire un algorithme qui permet de :

- Remplir le tableau TAB.
- Vérifier et afficher si le tableau est trié ou non dans l'ordre **décroissant**.
- Si le tableau **n'est pas trié** :
 - Déterminer et afficher le plus grand nombre **impair** s'il existe.
 - La moyenne des nombres **négatifs** s'ils existent.

Remarques :

- Le tableau TAB peut ne contenir aucun nombre impair. Dans ce cas, le message à afficher est : "Aucun nombre impair existe dans le tableau TAB".
- Le tableau TAB peut ne contenir aucun nombre négatif. Dans ce cas, le message à afficher est : "Aucun nombre négatif existe dans le tableau TAB".

Corrigé Exercice 1 :

Algorithme exo_01 ;

var TAB : tableau [1..150] entier ;

N, i, Somme, Nbre_neg, Max_impair : entier ;

Début

répéter

lire(N) ;

Jusqu'à ((N > 0) et (N ≤ 150)) ;

Pour i de 1 à N Faire

Ecrire ("TAB[" , i , "] = ") ;

Lire(TAB[i]) ;

FPour

/* vérification si le TAB est trié ou non */

i ← 1 ;

Tant Que ((i < N) et (TAB[i] ≥ TAB[i+1])) Faire

i ← i+1 ;

FTQue

Si (i=N) Alors

Ecrire (" Le Tableau TAB est trié ") ;

Sinon

Ecrire (" Le Tableau TAB n'est pas trié ") ;

/* Recherche le plus grand nombre impair s'il existe */

i ← 1 ;

Tant Que ((i ≤ N) et (TAB[i] mod 2 = 0)) Faire

i ← i+1 ;

FTQue

Si (i=N+1) Alors

Ecrire ("Aucun nombre impair existe dans le tableau TAB") ;

Sinon

Max_impair ← TAB[i] ;

i ← i+1 ;

Tant Que (i ≤ N) Faire

Si ((TAB[i] mod 2 = 1) et (TAB[i] > Max_impair)) Alors

Max_impair ← TAB[i] ;

Fsi

i ← i+1 ;

FTQue

Ecrire ("Le plus grand nombre impair est : " , Max_impair) ;

FSi

Corrigé d'Examen Final : Sujet -B-

/* Calcul de la moyenne des nombres négatifs s'ils existent */

```

i ← 1 ;
Tant Que ((i ≤ N) et (TAB[i] > 0)) Faire 0,50
    i ← i + 1 ;
FTQue

Si (i = N + 1) Alors
    Ecrire("Aucun nombre négatif existe dans le tableau TAB") ; 0,50
Sinon
    Somme ← TAB[i] ; 0,50
    Nbre_neg ← 1 ;
    i ← i + 1 ;
    Tant Que (i ≤ N) Faire 0,25
        Si (TAB[i] ≤ 0) Alors 0,25
            Somme ← Somme + TAB[i] ; 0,25
            Nbre_neg ← Nbre_neg + 1 ; 0,25
        Fsi
        i ← i + 1 ;
    FTQue
    Ecrire("La moyenne des nombres négatifs est : ", Somme/Nbre_neg) ; 0,50
FSi
FSi
Fin.
```

Exercice 2 :

Soit MAT une matrice N*M (N ≤ 25 et M ≤ 10) d'entiers.

Ecrire un algorithme qui permet de :

- Remplir la matrice MAT par des valeurs strictement **négatives**.
- Déterminer l'intervalle des valeurs de la matrice MAT (sous forme : [borne_inf...borne_sup]).
- Déterminer et afficher le **nombre de valeurs inférieurs** à la valeur de la moyenne de la matrice.

Corrigé Exercice 2 :

11 points

Algorithme exo_02;

```

var MAT: tableau [1..25, 1..10] entier ;
    N, M, i, j, Somme, Moy, borne_inf, borne_sup, Nbre_Inf_Moy: entier ; 0,50
```

Début

/* Remplir la matrice MAT par des valeurs strictement négatives.*/

Répéter

```

    Ecrire("Donner le nombre de lignes N = ") ; 0,25
    lire(N) ;
    Ecrire("Donner le nombre de colonnes M = ") ; 0,25
    lire(M) ;
    Jusqu'à ((N > 0) et (N ≤ 25)) et ((M > 0) et (M ≤ 10)) ; 0,25
    Pour i de 1 à N Faire 0,25
        Pour j de 1 à M Faire 0,25
            Répéter
                Ecrire("MAT[", i, ", ", j, "] = ") ;
                Lire(MAT[i, j]) ; 0,50
            Jusqu'à (MAT[i, j] < 0) ; 0,50
        FPour
    FPour
```

Corrigé d'Examen Final : Sujet -B-

/ Déterminer l'intervalle des valeurs de la matrice MAT (sous forme : [borne_inf...borne_sup]). */*

borne_inf ← MAT[1,1];

borne_sup ← MAT[1,1];

Pour i de 1 à N Faire

Pour j de 1 à M Faire

Si (borne_inf > MAT[i,j]) **Alors**
borne_inf ← MAT[i,j];

Fsi

Si (borne_sup < MAT[i,j]) **Alors**
borne_sup ← MAT[i,j];

Fsi

FPour

FPour

Écrire ("L'intervalle des valeurs de la matrice MAT est : [", borne_inf, "..", borne_sup, "]");

/ Déterminer et afficher le nombre de valeurs inférieures à la valeur de la moyenne de la matrice */*

Somme ← 0;

Pour i de 1 à N Faire

Pour j de 1 à M Faire

Somme ← Somme + MAT[i,j];

FPour

FPour

Moy ← Somme / (N * M);

Nbre_Inf_Moy ← 0;

Pour i de 1 à N Faire

Pour j de 1 à M Faire

Si (MAT[i,j] < Moy) **Alors**
Nbre_Inf_Moy ← Nbre_Inf_Moy + 1;

Fsi

FPour

FPour

Écrire ("Le nombre de valeurs inférieures à la valeur de la moyenne de la matrice est :", Nbre_Inf_Moy);

Fin.