IUT Arles– Info 1ère année – Matière AP (Algorithmique)	TD 12 Algorithmique

Exercice I : Quels résultats fourniront les programmes suivants :

```
Tableau x[2, 3] : numérique
Variables i, j, Val : numériques
Val ← 1
Répéter pour i = 1 à 2
     Répéter pour j = 1 à 3
          x[i, j] ← Val
          Val ← Val + 1
     Fin Pour
Fin Pour
Répéter pour j = 1 à 3
     Répéter pour i = 1 à 2
          Ecrire x[i,j]
     Fin Pour
Fin Pour
```

On va stocker en mémoire les valeurs suivantes :

	j=1	j=2	j=3
i=1	1	2	3
i=2	4	5	6

A l'affichage, on obtiendra les valeurs dans l'ordre suivant :
X[1,1] = 1
X[2,1] = 4
X[1,2] = 2
X[2,2] = 5
X[1,3] = 3
X[2,3] = 6

```
Tableau x[2, 3] : numérique
Variables i, j, Val : numériques
Val ← 1
Répéter pour i = 1 à 2
     Répéter pour j = 1 à 3
          x[i, j] ← Val
          Val ← Val + 1
     Fin Pour
Fin Pour
Répéter pour i = 1 à 2
     Répéter pour j = 1 à 3
          Ecrire x[i,j]
     Fin Pour
Fin Pour
```

On va stocker en mémoire les valeurs suivantes :

	j=1	j=2	j=3
i=1	1	2	3
i=2	4	5	6

A l'affichage, on obtiendra les valeurs dans l'ordre suivant :
X[1,1] = 1
X[2,1] = 2
X[1,2] = 3
X[2,2] = 4
X[1,3] = 5
X[2,3] = 6

Exercice II : Quels résultats fournira ce programme

```
Tableau t[4,2] : numérique
Variables k, m : numériques
Répéter pour k = 1 à 4
        Répéter pour m = 1 à 2
                t[k, m] ← k + m
        Fin Pour
Fin Pour
Répéter pour k = 1 à 4
        Répéter pour m = 1 à 2
                Ecrire t[k,m]
        Fin Pour
Fin Pour
```

La première boucle avec compteur remplit le tableau t :

2	3
3	4
4	5
5	6

La seconde boucle avec compteur en écrit les valeurs. D'où le résultat :

```
2
3
3
4
4
5
5
6
```

Exercice III : Même question en remplaçant la ligne t[k,m] ← k + m

Par :

a- t[k,m] ← 2 *(k-1) + m

1	2
3	4
5	6
7	8

D'où le résultat :

```
1
2
3
4
5
6
7
8
```

b- t[k,m] ← k + 4*(m-1)

1	5
2	6
3	7
4	8

D'où le résultat :

```
1
5
```

2 6 3 7 4 8

Exercice IV : Ecrire les instructions permettant d'obtenir la somme (Spos) des éléments positifs et la somme (Sneg) des éléments négatifs d'un tableau contenant 20 lignes et 50 colonnes.

```
Variables i, j, spos, sneg : numériques
Tableau T [20,50] : numérique
…
sneg ← 0
spos ← 0
Répéter pour i = 1 à 20
      Répéter pour j = 1 à 50
            Si T[i,j]>0 alors
                  spos ← spos + T[i,j]
            Sinon
                  sneg ← sneg + T[i,j]
            Fsi
      FinPour
FinPour
Ecrire « Somme des positifs : », spos
Ecrire « Somme des négatifs : », sneg
```

Exercice V : Ecrire les instructions permettant de compter le nombre de valeurs strictement positives d'un tableau contenant 20 lignes et 50 colonnes.

```
Variables i, j, npos : numériques
Tableau T [20,50] : numérique
…
npos ← 0
Répéter pour i = 1 à 20
      Répéter pour j = 1 à 50
            Si T[i,j]>0 alors
                  npos ← npos + 1
            Fsi
      FinPour
FinPour
Ecrire « Nombre de valeurs positives : », npos
```

Exercice VI : Ecrire les instructions permettant de déterminer la position du plus grand élément d'un tableau t. Pour cela, on placera dans les variables Imax et Jmax les valeurs correspondantes du premier et du second indice.

```
Variables i, j, max, Imax, Jmax : numériques
Tableau T [20,50] : numérique
…
Max ← T[1,1]
Imax ← 1
Jmax ← 1
Répéter pour i = 1 à 20
      Répéter pour j = 1 à 50
            Si T[i,j]>0 alors
                  Max ← T[i,j]
                  Imax ← i
                  Jmax ← j
            Fsi
      FinPour
FinPour
```

```
Ecrire « Le plus grand élément de t est t[», Imax, « , », Jmax,
« ]= », Max
```

Rmq : la valeur de Max peut être retrouvée en recherchant T[Imax, Jmax]

Exercice VII : Un tableau T contient N nombres. Calculer leur somme S, leur moyenne M et leur variance :

$$V = \frac{1}{N}\sum_{i=1}^{N}(Ti - M)^2$$

Etait il nécessaire de conserver les nombres dans un tableau pour effectuer tous ces calculs ?

```
Variables i, S, V, M : numériques
Tableau T[N] : numériques
S ← 0
Répéter pour i = 1 à N
      S ← S + T[i]
FinPour
M ← S/ N

// variance
V ← 0
Répéter pour i = 1 à N
      V ← V + (T[i] – M)* (T[i] – M)
FinPour
V ← V / N
Ecrire « Somme : », S
Ecrire « Moyenne », M
Ecrire « Variance », V
```

Avec cette formule, on utilise deux fois les valeurs contenues dans le tableau.

Exercice VIII : Un tableau T contient N nombres. Effectuer une permutation circulaire de ses valeurs : chaque élément doit prendre la valeur du suivant, sauf le dernier qui prend la valeur du premier.

Dans cet exercice comme pour l'échange des valeurs de deux variables, on utilise une variable auxiliaire pour conserver l'ancienne valeur de la première case de T :

```
Variable i, Aux : numériques
Tableau T [N]
// On suppose que T à reçu ses N valeurs
Aux ← T[1]
Répéter pour i =1 à N-1
     T[i] ← T[i+1]
Fin Pour
T[N] ← Aux
// la dernière case reçoit l'ancienne valeur de la première
```

Exercice IX : Un tableau T contient N nombres. Inverser l'ordre de ses éléments. Par exemple :

1	2	13	12	4	5

Devient

5	4	12	13	2	1

En écrivant cet algorithme pour la première fois, bien des programmeurs ont décidé d'échanger la première et la dernière case, puis la seconde avec l'avant dernière, …. , enfin la dernière et la première. En le faisant exécuter par un ordinateur, ils ont eu l'impression que l'algorithme n'avait rien fait, puis ils ont compris qu'ils avaient échangé deux fois les cases, remettant le tableau dans l'état initial.

Pour ne pas se soucier de la parité de N, il est très simple d'utiliser deux indices qui se déplacent, allant à la rencontre l'un de l'autre :

Variables IG, ID, Aux : numériques
Tableau T[N] : numérique

// on suppose les valeurs affectées à T
IG ← 1 // indice de l'élément de gauche
ID ← N // indice de l'élément de droite
Tant que IG < ID faire
    // Echanger T[IG] et T[ID]
    Aux ← T[IG]
    T[IG] ← T[ID]
    T[ID] ← Aux
    // Modifier les indices pour avancer vers le milieu du tableau
    IG ← IG + 1
    ID ← ID - 1
FinTantQue
Il est possible de n'utiliser qu'un seul indice en remarquant que ID + IG = N + 1 puis en remplaçant ID par N + 1 – IG, mais c'est un plaisir de matheux, qui ne rend pas l'algorithme plus clair, bien au contraire.

Exercice X : Un tableau T, de NE lignes et ND colonnes, contient les notes (de 0 à 20) de NE élèves à une série de ND devoirs. Chaque élève a eu une note à chaque devoir. On demande de calculer les distributions marginales, c'est à dire de remplir deux tableaux de nombres réels : l'un de NE éléments contiendra les moyennes aux différents élèves, l'autre de ND éléments contiendra les moyennes aux différents devoirs. On calculera aussi la moyenne générale (moyenne de toutes les notes).

Constantes NEMax = 40, NDMax = 20
Variables NE, ND, E, D, N, NN : numériques
Tableau Notes[NEMAX+1][NDMAX+1] : numérique

// on suppose que les notes ont été rentrées.
// Notes[E, D] désigne la note de l'élève n°E au devoir n°D
//Pour éviter de multiplier les tableaux, on choisit de ranger les distributions marginales pour //les lignes dans la colonne ND + 1 et pour les colonnes dans NE + 1. Le total, puis la
// moyenne générale sont stockés dans Notes[NE+1,ND+1]

Exercice XI : Un tableau T contient N nombres, tous compris entre 1 et 100. Vérifier qu'aucun nombre n'est présent deux fois dans le tableau.
La solution la plus simple à programmer est certainement la suivante :

```
Variable i, j, R : numériques
Tableau T [N] : numérique
R← 0 // nombre de répétition rencontrées
Répéter pour i = 1 à N-1
      Pour j = i+1 à N
            Si ( T[i] = T[j] et ( i # j )) alors R ← R + 1
            Fsi
      Finpour
FinPour
Si R = 0
      alors Ecrire « Tous les éléments sont distincts »
Sinon
      Ecrire « Au moins une valeur est répétée »
Fsi
```

Cet algorithme effectue toutes les comparaisons de paires d'éléments de T, soit (N-1)(N-2)/2 comparaisons.

Il est avantageux de remplacer les boucles Répéter pour par des boucles Tant Que, pour arrêter les comparaisons dès la rencontre d'une égalité. La variable R ne prenant que les valeurs 0 ou 1, on utilisera en programmation une variable booléenne.

```
TousDifférents  ← 1
i ← 1
Tant que (TousDifférents = 1 et i < N) faire
      j ← i +1
      Tant que (TousDifférents = 1 et j ≤ N) faire
            Si T[j] = T[i] alors
                  TousDifférents ← 0
            Sinon
                  j ← j + 1
            Fsi
      FinTantque
      i ← i +1
FinTantQue
Si TousDifférents = 1 alors
      Ecrire « Tous les éléments sont distincts »
Sinon
      Ecrire « Au moins une valeur est répétée »
Fsi
```