



DATE : 10/05/2016

NOM :

PRENOM :

Contrôle N° 2

(Durée : 2h)

Exercice 1 : (4 pts)

Donner le résultat de l'algorithme suivant.

Algorithme Exercice_1

Variables T : Tableau[6, 3] d' Entier

k, m : Entier

Début

Pour k ← 0 à 5 **Faire**

Pour m ← 0 à 2 **Faire**

 T[k, m] ← (k+1) * (m-1)

FinPour

FinPour

Pour k ← 0 à 2 **Faire**

Pour m ← 2 à 4 **Faire**

Ecrire(T(m, k))

FinPour

Ecrire('n')

FinPour

Fin



Exercice 2 : Somme de deux tableaux(4 pts)

Ecrire un algorithme qui réserve un tableau de dimension $N_{max} = 1000$. En prenant de l'utilisateur le nombre d'éléments à remplir, ce tableau est rempli à partir de deux tableaux de même longueur préalablement saisi par l'utilisateur. Le nouveau tableau sera la somme des éléments des deux tableaux de départ.

Tableau 1 :

2	5	4	6	8	23	3	1	4	9
---	---	---	---	---	----	---	---	---	---

Tableau 2 :

9	4	6	3	3	3	2	78	9	0
---	---	---	---	---	---	---	----	---	---

Résultat :

11	9	10	9	11	26	5	79	13	9
----	---	----	---	----	----	---	----	----	---

Exercice 3 : Recherche de la position du maximum d'un tableau 2D (4 pts)

Soit un tableau de deux dimensions $T[100,100]$ préalablement rempli d'entiers. Ecrire un algorithme qui recherche et affiche la position i et j de la valeur maximale du tableau. L'algorithme affiche aussi cette valeur.

Exercice 4 : Recherche de la première et la dernière occurrence (4 pts)

Ecrire un algorithme qui réserve un tableau de dimension $N_{max} = 1000$, puis le remplit d'entiers tout en prenant de l'utilisateur le nombre d'éléments à remplir et en vérifiant que cette valeur est inférieure à N_{max} .

Une fois le remplissage est terminé, le programme doit chercher, parmi les éléments introduits, la **position** de la **première** et de la **dernière occurrence** d'une valeur entière X donnée par l'utilisateur. Si l'élément X ne figure pas dans le tableau on doit le signaler à l'utilisateur à travers un message.



Exercice 5 : (4 pts)

On suppose qu'un tableau T de taille $N_{max} = 1000$ est rempli de N éléments entiers. Ecrire un algorithme qui met dans T[i] la valeur $T[i] + T[i + 1]$ si i est impair et la valeur $T[i] - T[i - 1]$ si i est pair, avec T[0] inchangé. Votre algorithme doit utiliser dans la soustraction la valeur originale de l'élément T[i - 1].

Tableau T :

2	5	4	6	8	23	3	1	4	9
---	---	---	---	---	----	---	---	---	---

Résultat T:

2	9	-1	14	2	26	-20	5	3	9
---	---	----	----	---	----	-----	---	---	---