



TD-TP N°2 : Tableau à N dimensions

Exercice 0. Somme des éléments de la matrice.

Ecrire un programme qui lit les dimensions L et C d'un tableau T à deux dimensions du type **Int** (exemple dimensions maximales: 50 lignes et 50 colonnes). Remplir le tableau par des valeurs entrées au clavier et afficher le tableau ainsi que la somme de tous ses éléments.

Exercice 1. Mise à zéro de la diagonale principale d'une matrice

Écrivez un programme en C qui demande à un utilisateur de remplir une matrice de N lignes et de N colonnes, puis le programme mettra à zéro la diagonale principale de cette matrice. Attention il faudra que les lignes et les colonnes soient égales pour la mise à zéro des diagonales.

Exercice 2. Matrice unitaire

Que fait ce programme ? Expliquez et faire une démonstration manuelle de ce programme

➤ **SOLUTION**

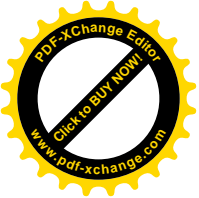
```
Int i, j, n, U [50][50] ;

Printf ("saisir la dimension de la matrice carrée max.50 :");

Scanf ("%d", &n);
for (i=1;i<n;i++)
{
    for (j=1;j<n;j++)
    {
        If ( i == j ) ;
        U [i] [j] = 1 ;
    }
    Else
        U [i] [j] = 0 ;
}

Printf (" le matrice unitaire de dimension %d : \n ,N ") ;

for (i=1;i<n;i++)
```



```
{  
    for (j=1;j<n;j++)  
  
    {  
  
        Printf (" 7% ", U [i] [j] ) ;  
  
        Printf (" \n ",) ;  
  
    }  
}
```

Exercice 3. Transposition d'une matrice

Écrivez un programme en C qui demande à un utilisateur de remplir une matrice de N lignes et de N colonnes, puis le programme fera le transposer de la matrice initiale. Attention La matrice transposée sera mémorisée dans une deuxième matrice B qui sera ensuite affichée.

Exercice 4. Multiplication d'une matrice par un son min

Écrivez un programme en C qui demande à un utilisateur de remplir une matrice de N lignes et de N colonnes, puis le programme recherchera d'abord le plus petit éléments de la matrice et par la suite il multipliera par le min. Attention Le résultat de la multiplication sera mémorisé dans une deuxième matrice A qui sera ensuite affichée.

Exercice 5. Addition de deux matrices

Écrivez un programme en C qui demande à un utilisateur de remplir une matrice de N lignes et de N colonnes, puis le programme affichera la somme de deux matrices. Attention Le résultat de l'addition sera mémorisé dans une troisième matrice C qui sera ensuite affichée

Exercice 6. Multiplication de deux matrices

Écrivez un programme en C qui demande à un utilisateur de remplir une matrice de N lignes et de N colonnes (matrice carré), puis le programme affichera le produit de deux matrices. Attention Le résultat du produit sera mémorisé dans une troisième matrice P qui sera ensuite affichée.

Y-a-t-il une autre possibilité de voir si les matrice n'est pas carré de faire le produit de ses deux matrice ??? Si oui écrire un programme qui va dans se sens ?

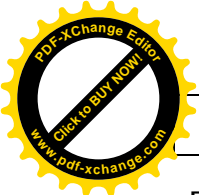
Exercice 7. Multiplication de deux matrices



Ecrire un programme qui transfère un tableau M à deux dimensions L et C (dimensions maximales: 10 lignes et 10 colonnes) dans un tableau V à une dimension L*C.

Exemple:

$$\begin{array}{|c|c|c|c|c|} \hline a & b & c & d & \\ \hline e & f & g & h & \\ \hline i & j & k & l & \\ \hline \end{array} ==> \begin{array}{|c|} \hline a & b & c & d & e & f & g & h & i & j & k & l & \\ \hline \end{array}$$



Exercice 1

Ecrire une procédure appelée *Produit de matrice* puis qui prend en paramètre d'entrée une matrice, ses dimensions et une valeur). Ensuite le programme affichera la multiplication d'une matrice A par un réel X.

- a) Le résultat de la multiplication sera mémorisé dans une deuxième matrice A qui sera ensuite affichée.
- b) Les éléments de la matrice A seront multipliés par X.

Exercice 2

Ecrire procédure appelé *diagonale* puis qui prend en paramètre d'entrée une matrice carrée et ses dimensions, ensuite le programme mettra à zéro les éléments de la diagonale principale d'une matrice *carrée* A donnée.

Exercice 3

Ecrire la fonction *SOMME_TAB* qui calcule la somme des N éléments d'un tableau TAB du type int. N et TAB sont fournis comme paramètres; la somme est retournée comme résultat du type long.

Exercice 3 bis

A l'aide des fonctions des exercices précédents, écrire un programme qui lit un tableau A d'une dimension inférieure ou égale à 100 et affiche le tableau et la somme des éléments du tableau.

Exercice 4

Ecrire la fonction *TRANSPO_MATRICE* à cinq paramètres MAT, L, LMAX, C, CMAX qui effectue la transposition de la matrice MAT en utilisant la fonction *PERMUTER*. *TRANSPO_MATRICE* retourne une valeur logique qui indique si les dimensions de la matrice sont telles que la transposition a pu être effectuée. Ecrire un petit programme qui teste la fonction *TRANSPO_MATRICE*.

Exercice 5

Ecrire une procédure *compte* qui demande à l'utilisateur de taper 10 entiers compris entre 0 et 20 qui seront stockés dans un tableau et qui affiche le nombre de fois qu'on a tapé un 0, le nombre de 1, le nombre de 2, ..., le nombre de 20.

Exercice 6

Ecrire une fonction appelé *tout* qui retourne une fois la moyenne quand l'utilisateur tape 1 et une autre fois le minimum quand l'utilisateur tape 2 et enfin le maximum quand l'utilisateur tape 3 d'un tableau unidimensionnels ;



TD-TP N° 3 : Trie d'un tableau

Exercice 1

Ecrivez un programme qui permette de saisir un nombre quelconque de valeurs, et qui les range au fur et à mesure dans un tableau. Le programme, une fois la saisie terminée, doit dire si les éléments du tableau sont tous consécutifs ou non.

Par exemple, si le tableau est :

12	13	14	15	16	17	18
----	----	----	----	----	----	----

ses éléments sont tous consécutifs. En revanche, si le tableau est :

9	10	11	15	16	17	18
---	----	----	----	----	----	----

ses éléments ne sont pas tous consécutifs.

Exercice 2

Ecrivez un programme qui trie un tableau dans l'ordre décroissant.

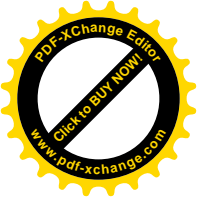
Vous écrirez bien entendu deux versions de cet algorithme, l'une employant le tri par insertion, l'autre le tri à bulles.

Exercice 3

Ecrivez un programme qui inverse l'ordre des éléments d'un tableau dont on suppose qu'il a été préalablement saisi (« les premiers seront les derniers... »)

Exercice 4

Ecrivez un programme qui permette à l'utilisateur de supprimer une valeur d'un tableau préalablement saisi. L'utilisateur donnera l'indice de la valeur qu'il souhaite supprimer. Attention, il ne s'agit pas de remettre une valeur à



zéro, mais bel et bien de la supprimer du tableau lui-même ! Si le tableau de départ était :

12	8	4	45	64	9	2
----	---	---	----	----	---	---

Et que l'utilisateur souhaite supprimer la valeur d'indice 4, le nouveau tableau sera :

12	8	4	45	9	2
----	---	---	----	---	---

Exercice 5

Ecrivez un programme qui recherche un élément saisi au clavier dans un dictionnaire. Le dictionnaire est supposé être codé dans un tableau préalablement rempli et trié.