

SÉRIE N° 16 : RÉVISION GÉNÉRALE**TABLEAUX-MATRICES-CHAÎNES DE CARACTÈRES****TABLEAUX :**

Dans cette partie on suppose avoir un tableau T d'entiers de taille n , déjà déclaré et initialisé

EXERCICE 1 :

Écrire un code qui affiche que les éléments impairs du tableau T .

EXERCICE 2 :

Écrire un code qui initialise chaque élément $T[i]$ d'un tableau T à la valeur 2^i .

EXERCICE 3 :

Écrire un code qui calcule le plus grand écart dans un tableau T de taille n (L'écart est la valeur absolue de la différence de deux éléments).

EXERCICE 4 :

Écrire un code qui insère un élément x à sa place dans un tableau qu'on supposera déjà trié.

Principe :

Ajouter la valeur de façon que le tableau reste trié, mais sans exécuter de tri.
Décaler les éléments en partant de la fin jusqu'à trouver une place correcte pour placer la nouvelle valeur ; il faut supposer que la place est suffisante c'est-à-dire que le nombre effectif d'éléments est strictement plus petit que la taille max du tableau en mémoire.

EXERCICE 5 :

Soit T un tableau d'entiers trié sauf la dernière case.

Écrire un code qui insère la valeur de la dernière case à sa place dans le tableau de façon que le tableau reste trié

Exemple :

Par exemple, soit T un tableau de 5 éléments : 1, 12, 15, 23, 10

Après la modification le tableau sera: 1, 10, 12, 15, 23

EXERCICE 6 :

Écrire un code qui permet de fusionner deux tableaux triés $T1$ de taille $N1$ et $T2$ de taille $N2$ dans un troisième tableau T .

Mettre bout à bout deux tableaux triés de façon que le tableau résultant soit trié, et ce, sans exécuter de tri (coûteux).

Indication : Parcourir les 2 tableaux (avec 2 indices), et recopier dans le 3ème tableau la plus petite des 2 valeurs du 1er et du 2nd tableau ; puis avancer dans le tableau d'où on a recopié la valeur. A la fin de ce traitement, l'un des tableaux est copié, il suffit de copier les éléments restant de l'autre.

EXERCICE 7 :

Soit t un tableau d'entiers non trié. Écrire un code qui affiche l'élément qui apparaît le plus souvent dans le tableau t , ainsi que son nombre d'occurrences. Si plusieurs éléments différents répondent au problème, votre code doit en afficher un, quel qu'il soit. Vous ne devez utiliser aucun autre tableau que celui sur lequel vous travaillez.

EXERCICE 8 :

Écrire un code qui affiche la valeur d'un polynôme P de degré n en un point x donné. Les coefficients $(a_n, a_{n-1}, \dots, a_0)$ de P sont contenus dans le tableau t ($t[0]=a_0, \dots, t[n]=a_n$)

$$p(x) = a_n * x^n + a_{n-1} * x^{n-1} + \dots + a_1 * x^1 + a_0$$

EXERCICE 9 :

Écrire un code qui permet de copier toutes les composantes strictement positives du tableau T dans un deuxième tableau T_{pos} et toutes les valeurs strictement négatives du tableau T dans un tableau T_{neg} . Afficher T_{pos} et T_{neg} .

EXERCICE 10 :

Écrire un code qui à partir d'un tableau d'entiers T , fournit le nombre de sous-séquences croissantes de ce tableau, ainsi que les indices de début et de fin de la plus grande sous-séquence.

Exemple :

Par exemple, soit T un tableau de 15 éléments :

1, 2, 5, 3, 12, 25, 13, 8, 4, 7, 24, 28, 32, 11, 14

- Les séquences strictement croissantes sont :
< 1, 2, 5 >; < 3, 12, 25 >; < 13 >; < 8 >; < 4, 7, 24, 28, 32 >; < 11, 14 >
- Le nombre de sous-séquence est : 6
- la plus grande sous-séquence est : < 4, 7, 24, 28, 32 >

EXERCICE 11 :

On met dans une variable **pivot** la valeur de la première case du tableau T .

On veut réordonner les éléments du tableau T en mettant en premier les éléments strictement plus petits que le pivot $T[i] < \text{pivot}$, puis les éléments égaux au pivot **pivot**, et en dernier les éléments strictement plus grands $T[i] > p$.

Exemple : Soit le tableau T suivant :

	0	1	2	3	4	5	6	7	8	9	10
T	8	2	5	8	1	34	21	6	9	14	3

- Sur le tableau exemple, en prenant comme valeur de pivot **pivot=8** on obtiendra le tableau résultat suivant :

	0	1	2	3	4	5	6	7	8	9	10
tab	3	2	5	1	6	8	8	21	34	9	14

- Notez que dans le résultat les nombres plus petits que le pivot 3; 2; 5; 1; 6 peuvent être dans n'importe quel ordre les uns par rapport aux autres.

Écrire un code qui permet de réordonner les éléments du T comme expliqué précédemment

MATRICES :

EXERCICE 1: Moyenne de notes par élève et par matière

On considère une matrice de notes dont les M lignes correspondent à M élèves et dont les N colonnes correspondent à N matières.

De plus on dispose d'un tableau **Coeff** dont les N valeurs correspondent à des coefficients.

- On suppose que la matrice **Notes** de taille $M \times N$ est déclarée et initialisée
- On suppose que le tableau **Coeff** de taille N est déclaré et initialisé

Exemple: Matrice Notes de taille 4x5

	0	1	2	3	4
0	10	11	9	7.5	15
1	11	10	3	19	8
2	10	10	10	10	10
3	2.5	15	6	19	4

Exemple: Le tableau Coeff de taille 5

	0	1	2	3	4
3	3	1	1	1	2

Ecrire un code qui permet de calculer et afficher la moyenne par élève

Exemple d'exécution :

la moyenne par eleve :

```
Moy de l'eleve 0 est 10.94
Moy de l'eleve 1 est 10.13
Moy de l'eleve 2 est 10.00
Moy de l'eleve 3 est 6.94
```

EXERCICE 2:

Ecrire un code qui permet de transférer une matrice M de taille $L \times C$ dans un tableau T de taille $L \times C$.

Afficher le tableau

EXERCICE 3:

Ecrire un code qui permet de rechercher d'existence d'un nombre donné x , dans une matrice M de taille $L \times C$.

EXERCICE 4: Les points cols d'un tableau

Les points cols d'un tableau M à deux dimensions (matrice) sont les éléments du tableau qui sont maximum sur leur ligne et minimum sur leur colonne.

Ecrivez un code qui recherche et affiche dans un tableau à deux dimensions les points cols.

CHAÎNES DE CARACTÈRES :

Dans cette partie on suppose avoir une chaîne de caractères **txt**, déjà déclarée et initialisée

EXERCICE 1: Sans utiliser string.h

Écrire un code qui affiche le nombre d'occurrences de la lettre 'f' dans la chaîne **txt**.

EXERCICE 2:

Écrire un code qui affiche la somme des codes ASCII des caractères d'une chaîne **txt**.

EXERCICE 3:

Écrire un code qui convertie une chaîne de caractères **txt** en majuscule

EXERCICE 4: sans utiliser string.h

Écrire un code qui permet de concaténer deux chaînes de caractères **txt1** et **txt2**.

Exemple : concaténer "bon" et "jour" donne "bonjour".

EXERCICE 5:

Écrire un code qui permet de supprimer les espaces supplémentaires (plus d'un espace) dans une chaîne de caractère **txt**.

EXERCICE 6: Anagramme

Deux mots sont des anagrammes si l'un est une permutation des lettres de l'autre. Par exemple les mots suivants sont des anagrammes :

- aimer et maire
- chien et niche

Par définition, on considère que deux mots vides sont des anagrammes.

Ecrire un code qui vérifie si deux mots **txt1** et **txt2** sont des anagrammes.

EXERCICE 7:

Écrire un code qui lit au clavier des caractères égaux à '0' ou '1' et calcule le nombre que ces chiffres représentent en binaire. On pourra faire afficher en base 10 le résultat.

Exemple :

Entrer un nombre binaire : 1001

La valeur en base 10 est :9