# Exercices supplémentaires sur les tableaux

Faites l’algorithme ou le programme des exercices suivants. (Les numéros de mes exercices préférés sont en gras). Vois les fichiers c# fournis.

## Tableau à une dimension

1. Faites une fonction qui **inverse les éléments d’un tableau** qui lui est passé en paramètre. Ce tableau est à une dimension. Interdit de prendre les fonctions sur les tableaux comme reverse.
2. Faites une fonction qui prend un tableau en paramètre. Il **cherche un élément** dans le tableau. Cet élément est aussi passé en paramètre. Votre fonction **retourne la position** dans le tableau de la première occurrence de cet élément. Par souci de performance, on vous demande de ne pas parcourir l’ensemble du tableau, mais bien d’arrêter dès que vous le trouvez. S’il ne le trouve pas retourner -1 comme position.
3. Soit un tableau de nombres (int[] tableau= new tableau[1000]), Créez un deuxième tableau de même dimension qui contiendra le **cumulatif** du premier. Faites une fonction qui construit le deuxième tableau soit le cumulatif. Votre fonction reçoit le premier tableau et retourne le deuxième tableau.

Exemple : Le tableau contient 2 10 5 7 …

Le cumulatif est 2 12 17 24 …

1. Faites une fonction qui permet de **compter** combien il y a eu de 0, de 1, de 2,... de 9 dans un string passé en paramètre. Servez-vous de la possibilité d’utiliser les indices sur les strings pour vous aider dans la lecture du string. Exemple : string nombre = txt.Text; on peut faire nombre[0] donc l’utiliser comme un tableau pour utiliser le premier chiffre de ce nombre. Vous utiliserez un tableau qui servira de compteur multiple pour faire le calcul. Ce dernier tableau sera retourné par la fonction. Vous pourrez l’afficher dans un listbox. Le tableau à l’indice 0 comptera le nombre de 0, à l’indice 1 le nombre de 1,…

Exemple : **Nombre :** 542783089211

**Voici le nombre : et son occurrence**

**0 : 1**

**1 : 2**

**2 : 2**

**3 : 1**

**4 : 1**

**5 : 1**

**6 : 0**

**7 : 1**

**8 : 2**

**9 : 1**

## Tableau à 2 dimensions :

Soit un tableau de 10 lignes et 3 colonnes. Les lignes représentent 10 étudiants et les colonnes représentent les notes de 3 exercices différents (même pondération).

Single[] tab = new single[10,3]

1. Faites une fonction qui me donne la **moyenne** d’un étudiant donné. Votre fonction prend le tableau en paramètre et le numéro de l’étudiant (numéro correspondant à sa ligne dans notre tableau), elle retourne la moyenne.
2. Faites une fonction qui retourne le nombre d’étudiants qui **ont une moyenne de leurs trois exercices** en haut de 80%
3. Quel est le **plus grand nombre** du tableau à deux dimensions passées en paramètre? Faites une fonction qui me retourne ce nombre et qui prend le tableau en paramètre.
4. Faites une fonction qui met les deux diagonales d’un tableau carré à zéro.

Exemple :

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Tableau de départ   |  |  |  |  | | --- | --- | --- | --- | | 2 | -4 | 3 | 5 | | 6 | 5 | 3 | 1 | | 8 | 7 | 3 | 5 | | 2 | 3 | 7 | -3 | | Tableau après la procédure   |  |  |  |  | | --- | --- | --- | --- | | 0 | -4 | 3 | 0 | | 6 | 0 | 0 | 1 | | 8 | 0 | 0 | 5 | | 0 | 3 | 7 | 0 | |