

TD 7

Manipulation de tableaux et de chaînes

Préambule concernant les chaînes de caractères

Les chaînes de caractères ont les mêmes propriétés qu'un tableau de caractères :

- la 1ère lettre se trouve en position 0 ;
- on accède à la lettre d'indice *i* dans la chaîne *mot* via l'opérateur [] : *mot[i]*

Les chaînes (pas les tableaux) disposent d'une méthode `size()` qui retourne le nombre de caractères de la chaîne. Ainsi, `mot.size()` retournera le nombre de caractères composant la chaîne *mot*.

Pour chaque exercice, vous préciserez si vous pouvez appliquer un des modèles d'algorithme vus en cours. Si c'est le cas, décrivez brièvement la stratégie qui permet d'adapter le modèle au problème posé.

Exercice 1 : Afficher un tableau

Proposez un algorithme qui permet d'afficher tous les entiers stockés dans un tableau composé de `nbCases`.

Exercice 2 : Incrémenter les valeurs négatives d'un tableau

Proposez un algorithme qui incrémente de 1 toutes les valeurs négatives d'un tableau d'entiers composé de `nbCases`.

Exercice 3 : Compter les valeurs négatives d'un tableau

Proposez un algorithme qui compte le nombre de valeurs négatives stockées dans un tableau d'entiers composé de nbCases.

Exercice 4 : Moyenne des valeurs d'un tableau

Proposez un algorithme qui calcule la moyenne de toutes les valeurs stockées dans un tableau d'entiers composé de nbCases.

Exercice 5 : Effectif d'un groupe de TP

On considère le type Etudiant et les tableaux etudiantsS1 et etudiantsS3 tels que définis ci-dessous :

```
struct Etudiant
{
    string nom ; // nom de l'étudiant
    string prenom; // prénom de l'étudiant
    unsigned short int td ; // numéro de TD de l'étudiant
    unsigned short int tp ; // numéro de TP de l'étudiant
};

// Nombre d'étudiants inscrits en semestre 1
const unsigned int EFFECTIF_S1 = 70 ;
// Nombre d'étudiants inscrits en semestre 3
const unsigned int EFFECTIF_S3 = 60 ;

// Etudiants inscrits en semestre 1 :
Etudiant etudiantsS1[EFFECTIF_S1];
// Etudiants inscrits en semestre 3 :
Etudiant etudiantsS3[EFFECTIF_S3];
```

Proposez un algorithme qui affiche combien d'étudiants sont inscrits dans un groupe de TP appartenant à une promotion donnée. On considérera que l'algorithme à élaborer aura en données :

- le tableau d'étudiants de la promotion à analyser ;
- le numéro du groupe de TP à analyser.

Exercice 6 : Inverser un tableau

Proposez un algorithme qui inverse les valeurs entières d'un tableau composé de nbCases : la 1 ère valeur passe en dernière position, la

dernière en 1ère position, la 2ème valeur passe en avant-dernière position, l'avant-dernière en 2ème position, etc.



Exercice 7 : Recherche de voyelles dans une chaîne de caractères

a/ Proposez un algorithme qui affiche la position de la première voyelle d'une chaîne de caractères.

b/ Proposez un algorithme qui affiche la position de la dernière voyelle d'une chaîne de caractères.

c/ Question optionnelle : Proposez un algorithme qui affiche la position de toutes les voyelles d'une chaîne de caractères.

Exercice 8 : Détection de palindromes

Un palindrome est un mot dont l'ordre des lettres reste le même qu'on le lise de gauche à droite ou de droite à gauche.

Exemples de palindromes : laval, radar, elle, kayak...

Proposez un algorithme qui détermine si une chaîne donnée est un palindrome ou pas. Pour simplifier le problème vous considérerez que la chaîne de caractères à analyser est toujours écrite en minuscules.

Exercice 9 : Affichage des tables de multiplication

On considère un tableau à deux dimensions représentant les tables de multiplication de 0 à 10 selon le format suivant :

	0	1	2	3	4	5	6	7	8	9	10
0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8	9	10
2	0	2	4	6	8	10	12	14	16	18	20
3	0	3	6	9	12	15	18	21	24	27	30
4	0	4	8	12	16	20	24	28	32	36	40
5	0	5	10	15	20	25	30	35	40	45	50
6	0	6	12	18	24	30	36	42	48	54	60
7	0	7	14	21	28	35	42	49	56	63	70
8	0	8	16	24	32	40	48	56	64	72	80
9	0	9	18	27	36	45	54	63	72	81	90
10	0	10	20	30	40	50	60	70	80	90	100

Proposez un algorithme qui considère un tel tableau en donnée ainsi que son nombre de lignes et de colonnes puis qui affiche toutes les tables de multiplication les unes en dessous des autres comme ci-dessous.

TABLE DE 0

$$0 \times 0 = 0$$

$$0 \times 1 = 0$$

$$0 \times 2 = 0$$

$$0 \times 3 = 0$$

$$0 \times 4 = 0$$

$$0 \times 5 = 0$$

$$0 \times 6 = 0$$

$$0 \times 7 = 0$$

$$0 \times 8 = 0$$

$$0 \times 9 = 0$$

$$0 \times 10 = 0$$

TABLE DE 1

$$1 \times 0 = 0$$

$$1 \times 1 = 1$$

$$1 \times 2 = 2$$

etc.

Bien sûr, votre algorithme ne devra réaliser **aucune multiplication** étant donné que tous les résultats sont déjà disponibles dans le tableau fourni en donnée...