

Fiche n°3 :
les tableaux à une dimension.

Ex 1. Soit T un tableau de n valeurs entières.

Ecrire un algorithme permettant d'**afficher à l'écran le contenu du tableau T.**

Jeu d'essai :

4	3	1	12	0	7	24	0	2	5
---	---	---	----	---	---	----	---	---	---

Résultat attendu : T < 4 3 1 12 0 7 24 0 2 5 >

Ex 2. Soit T un tableau de n valeurs entières.

Ecrire un algorithme permettant d'afficher à l'écran la **moyenne des valeurs contenues dans le tableau T.**

4	3	1	12	0	7	24	0	2	5
---	---	---	----	---	---	----	---	---	---

Résultat attendu : Moyenne des éléments de T : 5,8.

Ex 3. Soit T un tableau de n valeurs entières.

Ecrire un algorithme permettant de **calculer, puis afficher, le nombre d'apparitions d'une valeur x** dans le tableau T.

4	5	1	12	0	7	5	0	2	5
---	---	---	----	---	---	---	---	---	---

Valeur recherchée ? 5

Nombre d'apparitions : 3

Ex 4 : Ex 2. Soit T un tableau de n valeurs entières.

Ecrire un algorithme permettant d'**afficher à l'écran la plus grande des valeurs contenues** dans le tableau T.

4	5	1	12	0	7	5	0	2	5
---	---	---	----	---	---	---	---	---	---

Résultat attendu : Maximum des valeurs de T : 12

Ex 5. Soit un tableau contenant 12 notes (de 0 à 20) :

	0	1	2	3	4	5	6	7	8	9	10	11
lesNotes	11	12	11	10	6	9	15	10	15	14	17	2

Ecrire les structures itératives qui permettent :

- d'afficher toutes les notes,
- d'afficher la plus petite note,
- d'afficher toutes les notes dans l'ordre inverse du tableau,
- d'afficher le nombre de notes supérieures à 12,
- d'afficher la moyenne des notes supérieures à 10,
- d'afficher l'indice de la première case contenant la note 6,
- d'afficher l'indice de la dernière case contenant la note 10,
- d'afficher si oui ou non il y a un 20 dans les notes (en utilisant un booléen).

Ex 6. Conversion décimal → binaire sur 8 bits

Saisir un nombre décimal entier positif (le programme doit vérifier que le nombre saisi est bien entre 0 et max 255) et le convertir en binaire grâce à un tableau :

```

11  2
1   5  2
    1  2  2
      0  1  2
        1  0  2
          0  0  2
            0  0  2
              0  0  2
                0  0
  
```

1	1	0	1	0	0	0	0
---	---	---	---	---	---	---	---

Il reste à lire le tableau à l'envers : 00001011

Ex 7. Soit un tableau partiellement rempli, dont seulement les nVal premières valeurs ont été initialisées. La fin du tableau (entre l'indice nVal et le dernier indice) ne contient rien, ce sont des cases pour lesquelles on n'a pas encore fait d'affectation. Cela peut être le cas lors d'une saisie, par exemple.

On suppose en outre que, dans la partie initialisée du tableau, les entiers sont triés en ordre croissant.

—> **Ecrire un algorithme permettant de renvoyer l'indice où apparaît la première valeur supérieure à une valeur x donnée.**

Fiche n°3 : les tableaux à une dimension.


Jeux d'essai :

Cas n°1 : nombre de valeurs déjà saisies = 6, valeur à insérer : $x=9$.

Nombre de cases déjà remplies : 6

Valeur à inspecter (pour insertion éventuelle) : 9

1	3	5	8	14	20		
---	---	---	---	----	----	--	--



 valeur renvoyée : 4

Cas n°2 : nombre de valeurs déjà saisies = 6, valeur à insérer : $x=24$.

Nombre de cases déjà remplies : 6

Valeur à inspecter (pour insertion éventuelle) : 24

1	3	5	8	14	20		
---	---	---	---	----	----	--	--


 valeur renvoyée : 6

Ex 8. Ecrire l'algorithme qui **insère une valeur dans un tableau à un indice donné**, en décalant toutes celles à droite de cet indice.



Jeu d'essai n°1:

Nombre de cases déjà remplies : 6

Indice où aura lieu l'insertion : 4

Valeur à insérer : 9

1	3	5	8	9	14	20	
---	---	---	---	---	----	----	--



 valeurs décalées vers la droite


Jeu d'essai n°2 :

Nombre de cases déjà remplies : 6

Indice où aura lieu l'insertion : 6

Valeur à insérer : 24

1	3	5	8	14	20	24	
---	---	---	---	----	----	----	--


 valeur renvoyée : 6
 (aucune valeur à décaler)

Ex 9 : En vous servant des deux exercices précédents, écrire un algorithme permettant de **remplir un tableau (trié en ordre croissant) de nMax valeurs saisies au clavier dans un ordre quelconque.**

Principe général :

- on range la première valeur à l'indice zéro dans le tableau vide
- Pour chaque valeur suivante à insérer ;
 - On cherche où l'insérer (variable « posInsertion »)
 - On range la nouvelle valeur dans la case de numéro « posInsertion ».

Pendant la phase de mise au point, n affichera le tableau temporaire dès chaque saisie.

Ex 10 : Recherche dichotomique dans un tableau de N éléments trié en ordre croissant.

Principe :

On partage le tableau en deux sous-tableaux, d'indices [0, milieu] et [milieu+1, N-1]. On compare la valeur cherchée à la valeur " du milieu" du tableau, et suivant le résultat de la comparaison, on décide de continuer la recherche, selon le même principe dans l'un des deux sous-tableaux.

On répète le processus en divisant chaque fois la taille de la zone de recherche par deux, et on en arrive à un tableau de taille 1 : soit on a trouvé la valeur, soit elle n'est pas dans le tableau.

La fonction de recherche renvoie l'indice de la case où se trouve la valeur ou bien -1 si elle n'y est pas.

Exemple : recherche de la valeur 28.

1	3	5	8	9	14	20	28	32
---	---	---	---	---	----	----	----	----

↑

↑

Première phase :

- > position pointeur gauche : 0, position pointeur droite = 8
- > position du milieu : 4, valeur du milieu : 9
- > 28 > 9, donc on continue dans le sous-tableau de droite

1	3	5	8	9	14	20	28	32
---	---	---	---	---	----	----	----	----

↑

↑

Deuxième phase :

- > position pointeur gauche : 5, position pointeur droite = 8
- > position du milieu : 6, valeur du milieu : 20
- > 28 > 20, donc on continue dans le sous-tableau de droite

1	3	5	8	9	14	20	28	32
---	---	---	---	---	----	----	----	----

Troisième phase :

- > position pointeur gauche : 7, position pointeur droite = 8
- > position du milieu : 7, valeur du milieu : 28
- > 28 = valeur recherchée !