

Objectifs:

- Manipuler les tableaux à une seule dimension.
- Programmer toutes les variantes de recherche d'un élément dans un tableau.

Énoncé:

Créer des programmes en C qui permettent de chercher une valeur X (entré au clavier) dans un tableau d'entiers avec les possibilités suivantes:

1) Déterminer si l'entier X appartient au tableau T.

Exemple d'exécution 1:

```
Entrer la taille du tableau: 4
Remplissage du tableau:
Entrer l'element 1 :6
Entrer l'element 2 :3
Entrer l'element 3 :7
Entrer l'element 4 :55
Entrer la valeur a chercher: 7
La valeur 7 existe dans le tableau.
```

Exemple d'exécution 2:

```
Entrer la taille du tableau: 3
Remplissage du tableau:
Entrer l'element 1 :6
Entrer l'element 2 :3
Entrer l'element 3 :7
Entrer la valeur a chercher: 2
La valeur 2 n'existe pas dans le tableau.
```

2) Compter le nombre d'occurrence de l'entier X.

Exemple d'exécution :

```
Entrer la taille du tableau: 5
Remplissage du tableau:
Entrer l'element 1 :8
Entrer l'element 2 :5
Entrer l'element 3 :8
Entrer l'element 4 :4
Entrer l'element 5 :8
Entrer la valeur a chercher: 8
Le nombre d'occurrence de la valeur 8 est 3
```

3) Afficher la première position de l'entier X s'il existe ou le message "Élément n'existe pas" dans le cas contraire.

Exemple d'exécution :

```
Entrer la taille du tableau: 4
Remplissage du tableau:
Entrer l'element 1 :9
Entrer l'element 2 :7
Entrer l'element 3 :3
Entrer l'element 4 :7
Entrer la valeur a chercher: 7
La première position de la valeur 7 est: 2
```

4) Afficher la dernière position de l'entier X s'il existe ou le message "Élément n'existe pas" dans le cas contraire.

Exemple d'exécution :

```
Entrer la taille du tableau: 4
Remplissage du tableau:
Entrer l'element 1 :9
Entrer l'element 2 :7
Entrer l'element 3 :9
Entrer l'element 4 :2
Entrer la valeur a chercher: 9
La dernière position de la valeur 9 est: 3
```

5) Afficher toutes les positions de l'entier X s'il existe ou le message "Élément n'existe pas" dans le cas contraire.

Exemple d'exécution :

```
Entrer la taille du tableau: 5
Remplissage du tableau:
Entrer l'element 1 :3
Entrer l'element 2 :4
Entrer l'element 3 :3
Entrer l'element 4 :3
Entrer l'element 5 :2
Entrer la valeur a chercher: 3
Les positions de la valeur 3 sont:
1
3
4
```

Objectif

- Manipuler les boucles en langage C

Énoncé

L'objectif de cet exercice est de créer un programme en C qui permet de trouver la valeur inverse d'une factorielle.

Pour cela l'utilisateur saisie une valeur au clavier et le programme doit lui afficher la factorielle d'origine si elle existe ou le message "Aucune factorielle trouvée" sinon.

Il n'est pas demander de [calculer le factoriel d'un nombre](#) mais plutôt l'inverse.

Exemple 1 :

Donnée : 120

Résultat : 5!

Exemple 2 :

Donnée: 130

Résultat: Aucune factorielle trouvée

Indication

Penser à des divisions successives du nombre jusqu'à obtenir un quotient égal à 1.

Exemple pour la valeur 120:

$$120/1 = 120$$

$$120/2 = 60$$

$$60/3 = 20$$

$$20/4 = 5$$

$$5/5 = 1$$

Résultat = 5!

Objectif:

Parcourir et construire des tableaux

Énoncé:

Ecrire un programme qui lit la taille N d'un tableau T du type entier, remplit le tableau par des valeurs entrées au clavier et affiche le tableau.

Créer ensuite un tableau T1 contenant les entiers pairs de T et un tableau T2 contenant les entiers impaires de T, et afficher les deux tableaux T1 et T2.

Exemple d'exécution :

```
Entrer la taille du tableau : 6
Entrer l'element 1 :2
Entrer l'element 2 :3
Entrer l'element 3 :8
Entrer l'element 4 :4
Entrer l'element 5 :7
Entrer l'element 6 :10
Affichage du tableau saisi :
2 3 8 4 7 10
Affichage du tableau des nombres pairs :
2 8 4 10
Affichage du tableau des nombres impairs :
3 7
```