

# TP7 : les tableaux hérités du langage C

---

## Exercice 1

Dans le même projet :

Déclarer le tableau suivant dans le main

```
int Tableau[8] = {2,5,6,7,11,25,10,9}
```

2	5	6	7	11	25	10	9
---	---	---	---	----	----	----	---

- Ecrire et tester la fonction `void AfficheTab(int t[], int N)` qui permet d'afficher le contenu d'un tableau t de taille N

(main) Afficher le contenu de Tableau ;

- Ecrire et tester la fonction `void InitialiseTab(int t[], int N, int val)` qui permet d'initialiser tous les éléments d'un tableau t de taille N avec la valeur V.

(main) Initialiser toutes les cases de Tableau avec la valeur 0 ;

- Ecrire et tester la fonction `void ModifieTab(int t[], int N)` qui permet à l'utilisateur de modifier le contenu d'un élément du tableau t de taille N.
- Proposer un menu permettant de tester toutes les fonctions présentes dans ce sujet (pensez à utiliser un switch-case)

Exemple :

TP16 : Manipulation d'un tableau

1. Afficher le tableau
2. Initialiser le tableau
3. Modifier le tableau
4. Quitter l'application

## Exercice 2

Un IOT programmé en C enregistre la température mesurée par un capteur dans un tableau Temp.

Déclarer le tableau suivant dans le main

```
float Temp[10] = {12.5,12.8,9.5,12.9,13,12.7,18.3,12.75,13.0,12.6}
```

- Ecrire et tester la fonction `void AfficheTab(int t[], int N)` qui permet d'afficher le contenu d'un tableau t de taille N (voir exercice précédent)

(main) Afficher le contenu de Tableau ;

- Ecrire et tester la fonction `float Moyenne(int t[], int N)` qui retourne la valeur moyenne (la température moyenne) d'un tableau t de taille N .

(main) Appliquer la fonction Moyenne(...) sur le tableau Temp et afficher la température moyenne

- Ecrire et tester la fonction `void Tri_Bulle(int t[], int N)` qui permet de trier tous les éléments d'un tableau t de taille N dans l'ordre croissant.

Ressource : [http://lwh.free.fr/pages/algo/tri/tri\\_bulle.html](http://lwh.free.fr/pages/algo/tri/tri_bulle.html)

(main) Trier et afficher le contenu du tableau Temp ;

- Réaffecter au tableau Temp les valeurs initiales

```
float Temp[10] = {12.5,12.8,9.5,12.9,13,12.7,18.3,12.75,13.0,12.6}
```

- Ecrire et tester la fonction `float Mediane(int t[], int N)` qui retourne la valeur médiane d'un tableau t de taille N .

(main) Appliquer la fonction Mediane(...) sur le tableau Temp et afficher la valeur médiane.

## Annexe

L'acquisition de mesures de température (capteur pt100 ou autre) va mettre dans le tableau Temp une valeur toutes les minutes. Ces séries de mesures peuvent comporter des mesures incohérentes.

Après traitement, on ne conservera que la médiane (et non la moyenne) de ces séries de mesures. La valeur médiane (lire l'annexe 2) est la valeur qui se trouve au milieu d'un ensemble de nombres triés. Si cet ensemble contient un nombre pair de nombres, la médiane sera alors la moyenne des deux nombres du milieu.

L'utilisation de la médiane à la place de la moyenne est fréquente pour les mesures dans l'industrie.

Exemple, soit deux listes de mesures provenant d'un capteur sur une période de 1mn30s :

- L1 : 35,53°C, 35,23°C, 35,10°C, 35,02°C, 34,45°C
- L2 : 35,53°C, 35,23°C, 35,10°C, 34,45°C, 12,22°C

Dans la série L2, la mesure incohérente (12,22°C) serait pris en compte dans la moyenne et fausserait donc le résultat obtenu (la moyenne sans cette valeur est de 35,07°C contre 30,50°C si on en tient compte) :

Liste	L1	L2
Moyenne	35,066°C	30,506°C
Médiane	35,10°C	35,10°C

Ici, l'utilisation de la médiane comme technique de sélectivité permet d'atténuer ce type de problème.

L'utilisation de la valeur médiane est donc préférable à la valeur moyenne. Cependant, son utilisation implique le tri des données au préalable.