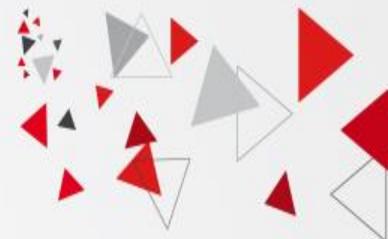


Les tableaux unidimensionnels



► Exemple introductif

Exemple:

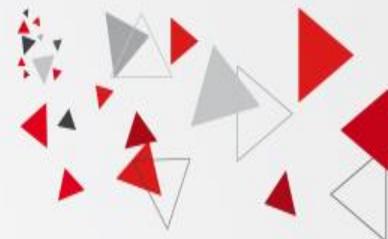
Nous désirons calculer les moyennes de 12 étudiants.

➤ **Solution 1:** Déclaration de 12 variables

$$\text{Moy} = (\text{N1} + \text{N2} + \text{N3} + \text{N4} + \text{N5} + \text{N6} + \text{N7} + \text{N8} + \text{N9} + \text{N10} + \text{N11} + \text{N12}) / 12$$

Inconvénients:

- Un nom pour chaque variable.
- Aucun lien entre les différentes variables.



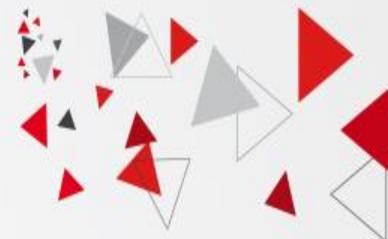
Solution:

Disposer d'un objet plus complexe, pour stocker ces moyennes, et y accéder à l'aide d'un indice.

Moyennes	15	12	10	9	13	20
Indices	0	1	2	3	4	5

A red arrow points from the text "Moyenne de l'élève n°2" to the index value "1" in the "Indices" row.

Moyenne de l'élève n°2



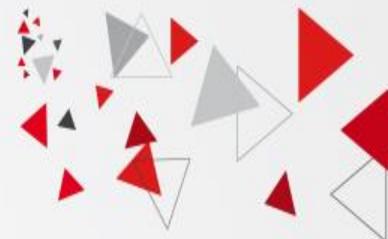
Définition

Un tableau est une structure de données permettant de stocker des données de même type.

Chacune des valeurs est repérée par un indice indiquant la position de la donnée dans le tableau.

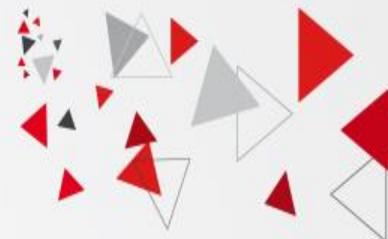
Un tableau est caractérisé par:

- ✓ Son nom
- ✓ Sa taille
- ✓ Sa dimension
- ✓ Le type de ses éléments



► Intérêts

- Gain de temps
- Rétrécissement du volume du programme
- Possibilité de réutilisation de toutes les valeurs ultérieurement dans le programme.



Déclaration d'un tableau

<Type_de_variable> <Nom_du_tableau> [<Taille>] ;

Exemple :

```
int TAB[10];  
float Notes[20];
```

- ✓ <Taille> est nécessairement une valeur numérique. Elle ne peut être en aucun cas une combinaison de variables du programme.
- ✓ Un élément du tableau est repéré par son indice. En langage C les tableaux commencent à partir de l'indice 0. L'indice maximum est donc N-1. (avec N : la taille du tableau)

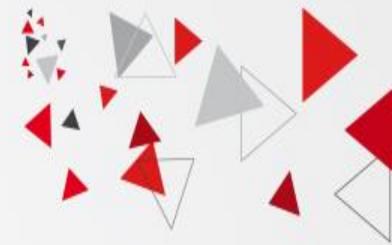
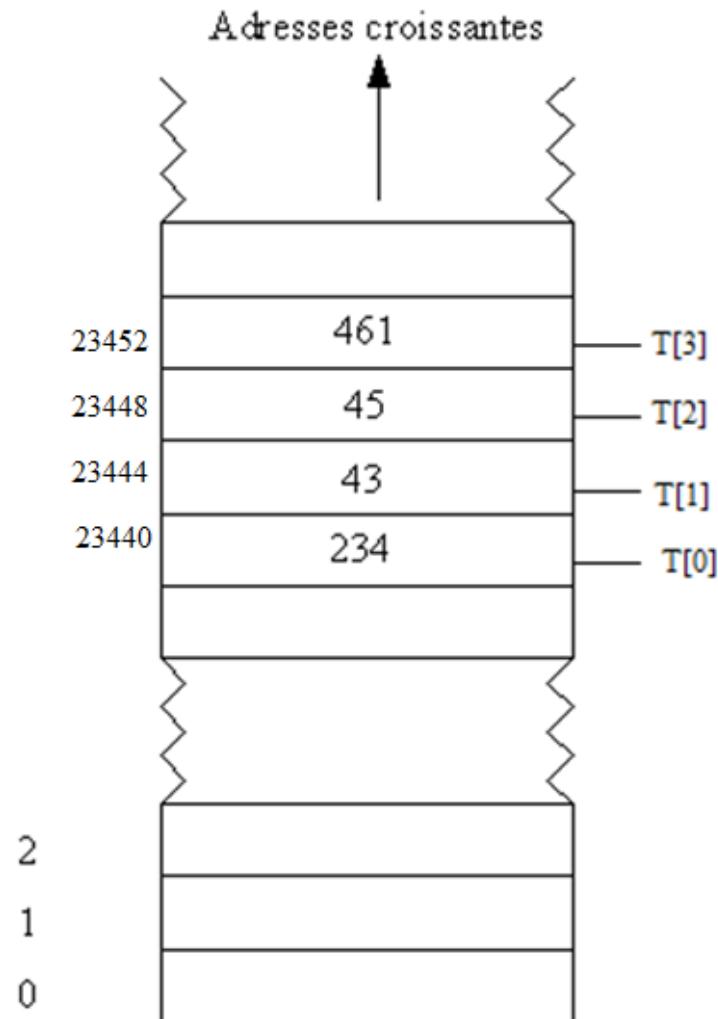


Tableau & mémoire (1/2)

Adresses

Mémoire

Variables



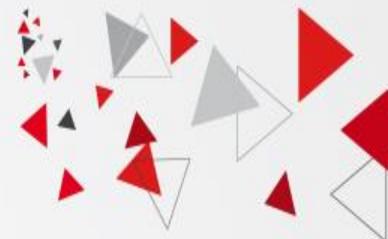
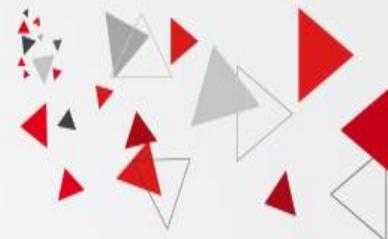


Tableau & mémoire (2/2)

- ✓ Pour un tableau de taille N, le compilateur C réserve N places en mémoire pour ranger les éléments du tableau.
- ✓ En langage C, le nom d'un tableau est le représentant de l'adresse du premier élément du tableau.
- ✓ Les adresses des autres composantes sont calculées (automatiquement) relativement à cette adresse.
- ✓ L'espace occupé en mémoire: Taille d'une composante en octets x N.



Taille maximale VS Taille réelle

- ✓ La déclaration suivante :

int TAB[10]; permet au compilateur de réserver un espace pour le tableau TAB avec une taille **maximale** de 10 * taille d'un entier.

- ✓ La taille réelle d'un tableau sera celle correspondante à l'élément référencé ayant le plus grand indice +1. La taille réelle doit être \leq à la taille maximale.

Exemple

TAB				Taille maximale = 10					
0	1	2	3						
2	7	1	8						

Taille Réelle = 4