

# INF155 — RAPPELS

## LES TABLEAUX 1D

Anis Boubaker, Ph.D.  
Maître d'enseignement  
École de Technologie Supérieure



# DÉCLARATION D'UN TABLEAU

## Syntaxe:

```
type identifiant[taille]= {valeur1, valeur2, ... };
```

## Exemple:

```
int notes[8]= {70, 89, 72, 65, 92, 77, 81, 78};
```

- L'initialisation est optionnelle;
- La taille doit être une valeur numérique ou une constante de précompilation

# ACCÈS AUX CASES D'UN TABLEAU

- Nous accédons aux cases d'un tableau comme aux variables normales : en indiquant l'identifiant du tableau
- La différence est que nous devons également préciser l'indice de la case.

```
int mon_tableau[10] = {0};  
mon_tableau[5] = 25; //Assigne la valeur 25 à la case 5  
//Accède à la case 5 du tableau et l'imprime à l'écran  
printf("La case 5 contient la valeur: %d", mon_tableau[5]);
```

- **Les indices d'un tableau commencent à 0.**

# TABLEAU EN PARAMÈTRE = PAR RÉFÉRENCE!

- Un tableau est **toujours** passé **par référence** en paramètre d'une fonction
- Un tableau en C est en fait un pointeur *masqué*...
- Si on modifie la case d'un tableau dans une fonction, le paramètre effectif (le tableau passé en paramètre) est modifié!

# TABLEAU EN PARAMÈTRE = PAR RÉFÉRENCE!

```
void incrementer(int tableau[], int nb_elements){  
    int i;  
    for( i=0 ; i < nb_elements ; i++){  
        tableau[i]++;  
    }  
}
```

```
int main(void){  
    int mon_tableau[5]={10,20,30,40,50};  
    int i;  
  
    incrementer( mon_tableau, 5);  
  
    for(int i=0; i<5; i++){  
        printf("Valeur %d : %d", i, mon_tableau[i]);  
    }  
}
```