



Les tableaux

10 | MAR | 2020

Dorian.H Mekni

Les tableaux

- Les tableaux sont des variables structurées.
- Les chaînes de caractères sont aussi déclarées sous forme de tableau.
- “Les tableaux sont une suite de variable de même type, situées dans un espace contigu en mémoire.
- Chaque valeur du tableau se voit attribué une case mémoire dans la RAM.

Déclaration de tableau

- On commence par le type d'abord.
- Par la suite défini le nombre de cases que l'on définit entre crochets
- Le premier index[i] d'un tableau se chiffre toujours à 0
- `int tableau[5]; // {0, 0, 0, 0, 0}`
- Si la valeur n'est pas précisée, chacune d'entre elle s'initialise automatiquement à 0.
- `int tableau[5] = {1} // {1, 0, 0, 0, 0}`
- Si seule la première valeur est désigné, les autres valeurs sont automatiquement initialisées à 0.

Initialisation de tableau

```
9  #include <stdio.h>
10 #include <stdlib.h>
11
12 int main(int argc, char *argv[])
13 {
14     int tableau[5];
15     int i = 0;
16
17     tableau[0] = 5;
18     tableau[1] = 7;
19     tableau[2] = 12;
20     tableau[3] = 24;
21     tableau[4] = 48;
22
23     for (i = 0 ; i < 4 ; i++)
24     {
25         printf("%d\n", tableau[i]);
26     }
27     return 0;
28 }
```



```
5
7
12
24
Program ended with exit code: 0
```

Tableau: avec CONST

```
#include <stdio.h>
#define SIZE_TAB 10

void show_tableau(int tab[], int size);

int main(void)
{
    int tableau[SIZE_TAB] = {16, 85, -2, 188, 3};

    show_tableau(tableau, SIZE_TAB);

    printf("\n\n");
    tableau[2] = -5;

    show_tableau(tableau, SIZE_TAB);
    printf("\n\n");

    return 0;
}

void show_tableau(int tab[], int size)
{
    int i;

    for (i = 0; i < SIZE_TAB; i++)
        printf("[%d] ", tab[i]);

}
```

Créer un tableau avec l'utilisation d'une fonction

```
28 #include <stdio.h>
29 #define SIZE_TAB 10
30
31 int *create_tableau(void);
32 void show_tableau(int tab[], int size);
33
34 /*-----*/
35
36 int main(void)
37 {
38     int *tableau = create_tableau();
39
40     show_tableau(tableau, SIZE_TAB);
41
42     printf("\n\n");
43     tableau[2] = -5;
44
45     show_tableau(tableau, SIZE_TAB);
46     printf("\n\n");
47
48     return 0;
49 }
50
51 /*-----*/
52
53 int *create_tableau(void)
54 {
55     static int tableau_int[5];
56     int i;
57
58     for(i = 0; i < 5; i++)
59         tableau_int[i] = i * 3;
60
61     return tableau_int;
62 }
63
64 /*-----*/
65
66 void show_tableau(int tab[], int size)
67 {
68     int i;
69
70     for (i = 0; i < SIZE_TAB; i++)
71         printf("[%d] ", tab[i]);
72
73 }
```

Créer un damier

```
#include <stdio.h>
#define Number_of_lines 3
#define Number_of_columns 2

int main(void)
{
    int damier[3][2] = { {1, 2}, {3, 4}, {5, 6}};
    int i, j;

    for(i = 0 ; i < Number_of_lines ; i++)
        for(j = 0 ; j < Number_of_columns ; j++)
            printf("Element [%d][%d] = %d\n", i , j, damier[i][j]);

    return 0;
}
```

```
Element [0][0] = 1
Element [0][1] = 2
Element [1][0] = 3
Element [1][1] = 4
Element [2][0] = 5
Element [2][1] = 6
Program ended with exit code: 0
```

