

A black level tool, resembling a small spirit level or a precision square, is positioned on a color calibration chart. The chart features a grid of squares in various shades of blue, green, and yellow. The tool has a circular lens on top and a rectangular base with a central cutout. The background is slightly blurred, showing more of the chart and a dark surface on the right.

Fonctions et tableaux

En paramètres

L'identificateur d'un tableau étant un pointeur sur le premier élément, le passage d'un tableau en argument d'une fonction ne peut se faire **que par adresse**.

 Cela implique qu'une fonction pourra toujours modifier le tableau envoyé.

Même si ce n'est pas obligatoire, il est **vivement recommandé** de passé la taille du tableau en argument pour pouvoir le parcourir et éviter les **erreurs de segmentation**.

Cas des tableaux

Prototype 1

```
#include <stdio.h>
```

```
// Déclaration
```

```
void modif_tableau (int *, int);
```

```
//Programme principal
```

```
int main () {
```

```
    int array[4] = {1, 2, 3, 5};
```

```
    modif_tableau(array, 4);
```

```
}
```

```
// Définition
```

```
void modif_tableau (int * t, int taille) {
```

```
    t[0] = 6;
```

```
    *(t+1) = 4;           // Élément suivant
```

```
    t[taille - 1] = 42    // Dernier élément du tableau
```

```
}
```

Cas des tableaux

Prototype 2

```
#include <stdio.h>
```

```
// Déclaration
```

```
void modif_tableau (int [], int);
```

```
//Programme principal
```

```
int main () {
```

```
    int array[4] = {1, 2, 3, 5};
```

```
    modif_tableau(array, 4);
```

```
}
```

```
// Définition
```

```
void modif_tableau (int t[], int taille) {
```

```
    t[0] = 6;
```

```
    *(t+1) = 4;           // Élément suivant
```

```
    t[taille - 1] = 42    // Dernier élément du tableau
```

```
}
```

Renvoyer un tableau

Il n'est pas possible de renvoyer un tableau créé dans une fonction via l'instruction `return` (on renverrait un pointeur vers une zone qui n'existe plus, une fois sorti de la fonction).

Par contre, le tableau étant passé par adresse, il est possible de le modifier dans son espace mémoire d'origine.

Pour "renvoyer un tableau", il faudra donc :

- 1) Déclarer un tableau en dehors de la fonction
- 2) Passer ce tableau à la fonction par adresse
- 3) Remplir/Modifier le tableau dans la fonction

"Renvoyer" un tableau

```
#include <stdio.h>
```

```
// Déclaration
```

```
void remplirAvec(int *, int, int);
```

```
//Programme principal
```

```
int main () {
```

```
    int array[4];
```

```
    remplirAvec(array, 4, 42);
```

```
    printf("%d", array[3]);    // 42 👍
```

```
}
```

```
// Définition
```

```
void remplirAvec (int * t, int taille , int nombre) {
```

```
    for (int i = 0; i < taille; i++) {
```

```
        t[i] = nombre;
```

```
    }
```

```
}
```