

Séance 9 : Les tableaux

Le langage C permet l'utilisation de tableaux. Un tableau simple (à une dimension) permet de stocker « n » valeurs **du même type** (`int`, `char`, `float`, `double`, etc.). Chaque élément de ce tableau est **indiqué de 0 à n – 1**.

Le langage C en outre ne s'arrête pas là et permet la manipulation de tableaux multidimensionnels comme nous allons le voir plus loin.

Déclaration

Un tableau en C contient « n » éléments du même type. Cette taille est fixée une fois pour toute lors de la déclaration. Par exemple :

```
int note[6];
```

Dans cet exemple, on déclare un tableau « `note` » de 6 entiers. La déclaration commence donc comme pour les variables par le type des éléments du tableau, de son nom, puis de sa taille entre crochets. De même pour des tableaux multidimensionnels :

```
double releve[100][200];
```

Ici, on déclare un tableau à deux dimensions contenant 20 000 éléments.

Il est possible comme pour les déclarations de type simple d'affecter une valeur aux éléments du tableau à sa déclaration, par exemple :

```
int note[6] = { 4, 12, 13, 6, 10, 17};  
double releve[2][2] = { { 4.0, 6.0}, { 7.0, 12.0}};
```

Cette déclaration se fait par l'énumération des éléments du tableau, séparés par des virgules et mis entre accolades. Le nombre d'éléments énumérés doit correspondre à la taille du tableau (dans le cas contraire le compilateur génère une erreur).

Ce type d'affectation ne peut se faire qu'à la déclaration.

Utilisation

On peut utiliser les éléments du tableau soit pour leur affecter une valeur, soit pour utiliser la valeur d'un élément :

```
note[4]    = 5;           /* affectation */
moyenne    = (note[0]+note[1]+note[2]+note[3]+note[4]+
               note[5])/6; /* utilisation */
```

Un élément du tableau s'utilise donc à peu près comme une variable et est constitué du nom du tableau et de l'indice de l'élément entre crochet. Cet indice peut être soit une valeur entière, soit une variable entière comme nous allons le voir par la suite.

Rappelons que les éléments d'un tableau sont indicés à partir de 0, le premier élément du tableau est donc l'élément d'indice 0 et le dernier élément celui d'indice « n-1 », n étant la taille du tableau.

Lorsque l'on effectue un traitement les éléments d'un tableau, on utilisera souvent la boucle « for ». Par exemple dans le cadre du calcul d'une moyenne :

```
somme      = 0;
for(i=0 ; i<6 ; i++)
{
    somme += note[i];
}
moyenne = somme/6;
```

où i est une variable entière.

Exercice 1

Ecrire un programme qui demande à l'utilisateur 5 notes (qui seront stockés dans un tableau d'éléments de type `double`). Puis affiche, la moyenne de ces notes.

Compléter ensuite ce programme pour qu'il affiche aussi la note la plus basse et la note la plus haute.

Exercice 2

Ecrire un programme qui affiche les 20 premières lignes d'un triangle de Pascal. Rappelons que la seule règle pour le construire est la suivante : chaque nombre est la somme des deux nombres voisins placés au dessus de lui.

Le programme affichera donc un résultat du type :

```
1
1  1
1  2  1
1  3  3  1
1  4  6  4  1
```

et ainsi de suite.

Exercice 3

Ecrire un programme permettant la saisie de deux matrices 3x3 (type entière). Puis proposer à l'utilisateur d'en faire la multiplication ou l'addition puis afficher le résultat.

Compléter ensuite le programme en y ajoutant le calcul du déterminant.