COMPUTER SCIENCE : C++

Arrays in C++

An array is a collection of similar items stored in contiguous memory locations. In programming, sometimes a simple variable is not enough to hold all the data. For example, lets say we want to store the marks of 500 students, having 500 different variables for this task is not feasible, we can define an array with size 500 that can hold the marks of all students.

Declaring an array

There are couple of ways to declare an array.

Method 1:

```
int arr[5];
arr[0] = 10;
arr[1] = 20;
arr[2] = 30;
arr[3] = 40;
arr[4] = 50;
```

Method 2:

```
int arr[] = {10, 20, 30, 40, 50};
```

Method 3:

```

int arr[5] = {10, 20, 30, 40, 50};
```

Accessing Array Elements

Array index starts with 0, which means the first array element is at index 0, second is at index 1 and so on. We can use this information to display the array elements. See the code below:

```
#include <iostream>
using namespace std;

int main(){
   int arr[] = {11, 22, 33, 44, 55};
   cout<<arr[0]<<endl;
   cout<<arr[1]<<endl;
   cout<<arr[2]<<endl;
   cout<<arr[3]<<endl;
   cout<<arr[4]<<endl;
   return 0;
}
```

Output:

```
11
22
33
44
55
```

Although this code worked fine, displaying all the elements of array like this is not recommended. When you want to access a particular array element then this is fine but if you want to display all the elements then you should use a loop like this:

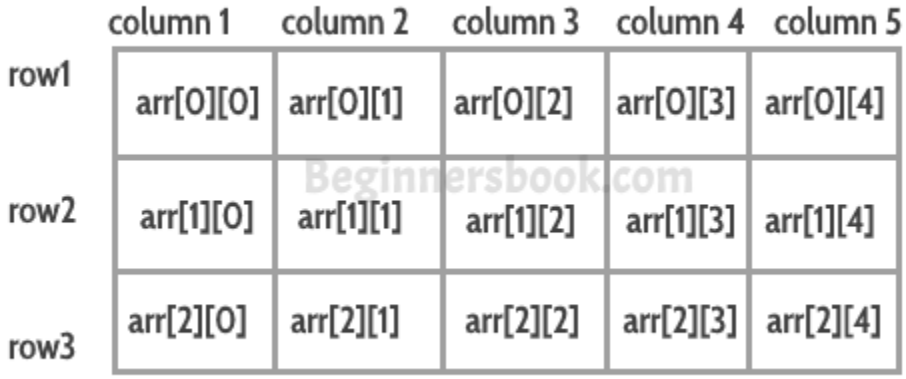
```
#include <iostream>
using namespace std;

int void main(){
   int arr[] = {11, 22, 33, 44, 55};
   int n=0;

   while(n<=4){
      cout<<arr[n]<<endl;
      n++;
   }
}
```

Multidimensional Arrays in C++

Multidimensional arrays are also known as **array of arrays**. The data in multidimensional array is stored in a tabular form as shown in the diagram below:



A two dimensional array:

```
int arr[2][3];
```

This array has total 2*3 = 6 elements.

A three dimensional array:

```
int arr[2][2][2];
```

This array has total 2*2*2 = 8 elements.

Two dimensional array

Lets see how to declare, initialize and access Two Dimensional Array elements.

How to declare a two dimensional array?

```
int myarray[2][3];
```

Initialization:

We can initialize the array in many ways:

Method 1:

```
int arr[2][3] = {10, 11 ,12 ,20 ,21 , 22};
```

Method 2:

This way of initializing is preferred as you can visualize the rows and columns here.

```
int arr[2][3] = {{10, 11 ,12} , {20 ,21 , 22}};
```

Accessing array elements:

arr[0][0] – first element

arr[0][1] – second element

arr[0][2] – third element

arr[1][0] – fourth element

arr[1][1] – fifth element

arr[1][2] – sixth element

Two dimensional array in C++

```
#include <iostream>
using namespace std;

void  main(){
   int arr[2][3] = {{11, 22, 33}, {44, 55, 66}};
   for(int i=0; i<2;i++){
      for(int j=0; j<3; j++){
        cout<<"arr["<<i<<"]["<<j<<"]:
"<<arr[i][j]<<endl;
      }
   }
};
```

Output:

```
arr[0][0]: 11
arr[0][1]: 22
arr[0][2]: 33
arr[1][0]: 44
arr[1][1]: 55
arr[1][2]: 66
```

Three dimensional array

Lets see how to declare, initialize and access Three Dimensional Array elements.

Declaring a three dimensional array:

```
int myarray[2][3][2];
```

Initialization:

We can initialize the array in many ways:

Method 1:

```
int arr[2][3][2] = {1, -1 ,2 ,-2 , 3 , -3, 4, -4, 5,
-5, 6, -6};
```

Method 2:

This way of initializing is preferred as you can visualize the rows and columns here.

```
int arr[2][3][2] = {
     { {1,-1}, {2, -2}, {3, -3}},
     { {4, -4}, {5, -5}, {6, -6}}
```

```
}
```

Three dimensional array

```
#include <iostream>
using namespace std;

int main(){
   // initializing the array
   int arr[2][3][2] = {
      { {1,-1}, {2,-2}, {3,-3} },
      { {4,-4}, {5,-5}, {6,-6} }
   };
   // displaying array values
   for (int x = 0; x < 2; x++) {
     for (int y = 0; y < 3; y++) {
       for (int z = 0; z < 2; z++) {
         cout<<arr[x][y][z]<<" ";
       }
     }
   }
   return 0;
}
```

Output:

```
1 -1 2 -2 3 -3 4 -4 5 -5 6 -6
```