3 Dimensional Arrays in C

We will try to understand 3 Dimensional Arrays in C in this class.

3 Dimensional Arrays in C The below shows the way the 3-dimensional arrays are declared in C.

Visualizing a 3 Dimensional array For example, if we consider our first declaration,

it states that it has four columns/Blocks of two-dimensional arrays, and each two-dimensional array has three rows and two columns.

The number of elements in the above array is 4\*3\*2 = 24.

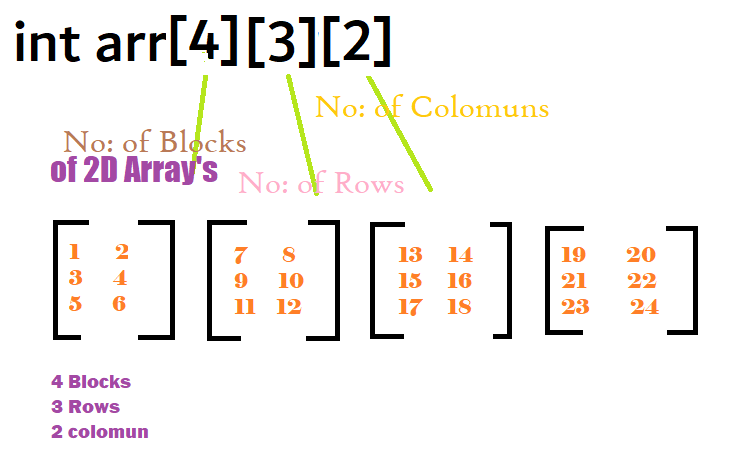
For example, assume that our first array has elements from 1 to 24.

The visualization of the 3-dimensional array is shown below.

Accessing the elements of the 3-dimensional arrays All the elements are stored linearly on memory.

If we want to access two(2), then the declaration should be a[0] [1] [2].

The above declaration means that in the zero blocks of the 2-dimensional arrays, access the element in the first row and second column.



#include<stdio.h>

int i,j,k; //variables for nested for loops

int main()

{

int arr[4][3][2]; //array declaration

printf("enter the values in the array: \n");

for(i=1;i<=4;i++) //represents block

{

for(j=1;j<=3;j++) //represents rows

{

for(k=1;k<=2;k++) //represents columns

{

printf("the value at arr[%d][%d][%d]: ",i,j,k);

scanf("%d",&arr[i][j][k]);

}

}

}

printf("printing the values in array: \n");

for(i=1;i<=4;i++)

{

for(j=1;j<=3;j++)

{

for(k=1;k<=2;k++)

{

printf("%d ",arr[i][j][k]);

if(k==2)

{

printf("\n");

}

}

}

printf("\n");

}

return 0;

}