

Multi-Dimensional Arrays

A multi-dimensional array can be termed as an array of arrays that stores homogeneous data in tabular form. Data in multidimensional arrays is generally stored in row-major order in the memory. We have 'n' number of indexes in this array. The requirement of the memory increases with the number of indices that it uses. These arrays are declared and initialized in the same manner as that of one and two-dimensional arrays.

The general form of declaring N-dimensional arrays is shown below.

Syntax:

type array_name[size1][size2]...[sizeN];

Here, type is the data type of the elements that will be stored in the array, array_name is the name of the array, and size1, size2, ..., sizeN are the sizes of each dimension of the array.

Example: int a[10]b[10]c[10]

Write a program to read and display a 2*2*2 array.

```
#include <stdio.h>

int main()
{
    int arr[3][3][3],i,j,k;
    printf("\n Enter the elements for the array:");
    for(i=0;i<2;i++)
    {
        for(j=0;j<2;j++)
        {
            for(k=0;k<2;k++)
            {
                printf("\n array [%d][%d][%d] = ",i,j,k);
                scanf("%d",&arr[i][j][k]);
            }
        }
    }
    printf("\n The matrix is:");
    for(i=0;i<2;i++)
    {
        for(j=0;j<2;j++)
        {
            printf("\n");
            for(k=0;k<2;k++)
                printf("%d ",arr[i][j][k]);
        }
    }
    return 0;
}
```

Output:

Enter the elements for the array:

array [0][0][0] = 1

array [0][0][1] = 2

array [0][1][0] = 3

array [0][1][1] = 4

array [1][0][0] = 5

array [1][0][1] = 6

array [1][1][0] = 7

array [1][1][1] = 8

The matrix is:

1 2

3 4

5 6

7 8