

Lecture 7

Arrays



Array

Array is a collection of homogenous data stored under unique name. The values in an array is called as 'elements of an array.' These elements are accessed by numbers called as 'subscripts or index numbers.' Arrays may be of any variable type. Array is also called as 'subscripted variable.'

Types of an Array :

One / Single Dimensional Array

Two Dimensional Array

Single / One Dimensional Array :

The array which is used to represent and store data in a linear form is called as 'single or one dimensional array.'

Syntax: <data-type> <array_name> [size];

Example:

```
int a[3] = {2, 3, 5};
```

```
char ch[20] = "TechnoExam" ;
```

```
float stax[3] = {5003.23, 1940.32, 123.20} ;
```

Total Size (in Bytes):

total size = length of array * size of data type

In above example, a is an array of type integer which has storage size of 3 elements. The total size would be 3

* 2 = 6 bytes.

* MEMORY ALLOCATION :

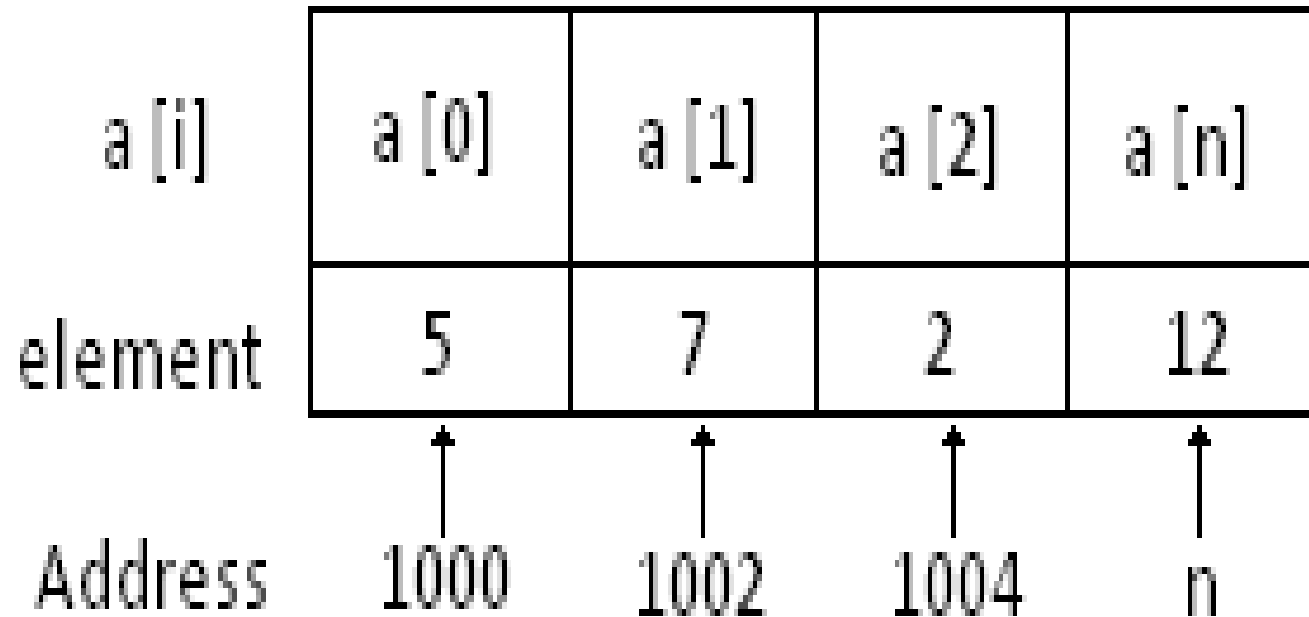


Fig : Memory allocation for one dimensional array

Single / One Dimensional Array :

Program :

```
#include <stdio.h>
#include <conio.h>
void main()
{
int a[3], i;
clrscr();
printf("\n\t Enter three numbers : ");
for(i=0; i<3; i++)
{
scanf("%d", &a[i]); // read array
}
printf("\n\n\t Numbers are : ");
for(i=0; i<3; i++)
{
printf("\t %d", a[i]); // print array
}
getch();}
```

Single / One Dimensional Array :

Program :

Output :

Enter three numbers : 9 4 6
Numbers are : 9 4 6_

Main Features of One Dimensional Array :

Array size should be positive number only.

String array always terminates with null character ('\0').

Array elements are countered from 0 to n-1.

Useful for multiple reading of elements (numbers).

Two Dimensional Array :

The array which is used to represent and store data in a tabular form is called as **'two dimensional array.'** Such type of array specially used to represent data in a matrix form.

The following syntax is used to represent two dimensional array.

Syntax:

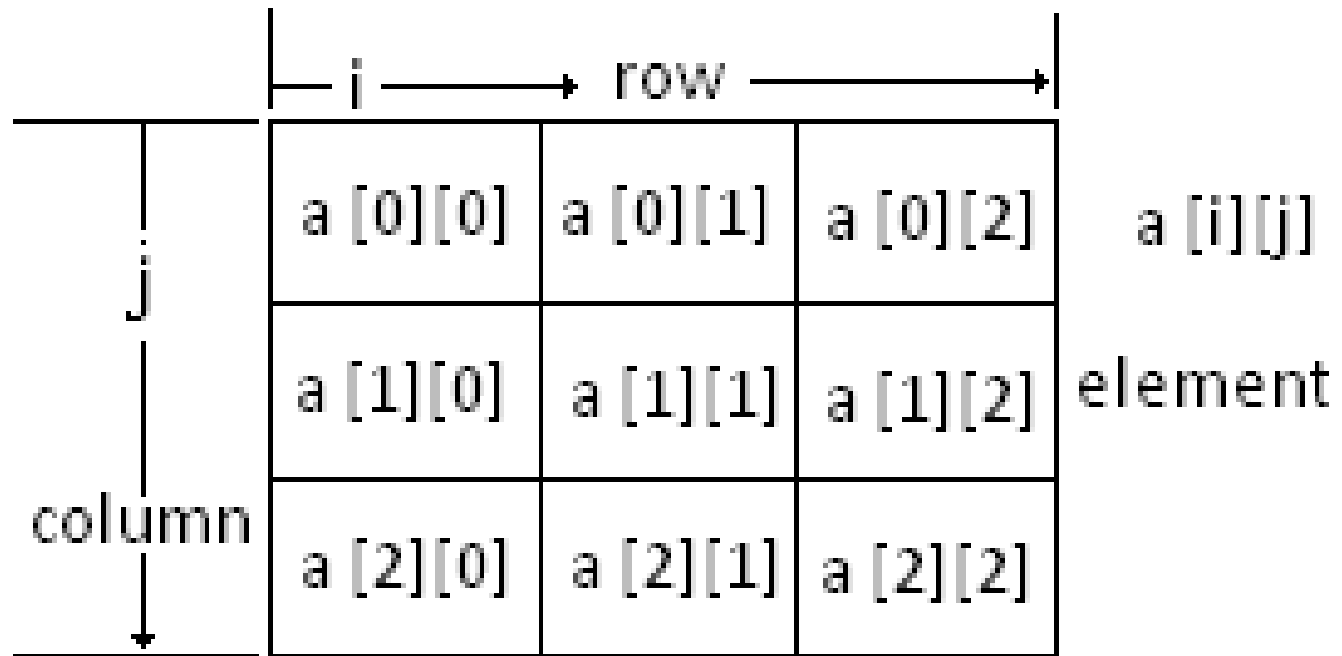
```
<data-type> <array_nm> [row_subscript][column-subscript];
```

Example:

```
int a[3][3];
```

In above example, a is an array of type integer which has storage size of 3 * 3 matrix. The total size would be $3 * 3 * 2 = 18$ bytes.

It is also called as 'multidimensional array.'



Memory allocation for two dimensional array


```
#include <stdio.h>
#include <conio.h>
void main()
{
    int a[3][3], i, j;
    clrscr();
    printf("\n\t Enter matrix of 3*3 : ");
    for(i=0; i<3; i++)
    {
        for(j=0; j<3; j++)
        {
            scanf("%d",&a[i][j]); //read 3*3 array
        }
    }
    printf("\n\t Matrix is : \n");
    for(i=0; i<3; i++)
    {
        for(j=0; j<3; j++)
        {
            printf("\t %d",a[i][j]); //print 3*3 array
        }
        printf("\n");
    }
    getch();
}
```

Program :

Output :

Enter matrix of 3*3 : 3 4 5 6 7 2 1 2 3

Matrix is :

3	4	5
6	7	2
1	2	3_