

Problem Solving and Programming using C (PSPC)

Dr. M. Raja Roy

Associate Professor

Anil Neerukonda Institute of Technology and Sciences

www.anits.edu.in

www.mrrtechnical.co.in

UNIT-III

Array

Dr. M. Raja Roy, Associate Professor, ANITS

www.mrrtechnical.co.in

Array

Array in C language is a collection or group of elements (data). All the elements of c array are homogeneous (similar). It has contiguous memory location.

C array is beneficial if you have to store similar elements. Suppose you have to store marks of 50 students, one way to do this is allotting 50 variables. So it will be typical and hard to manage. For example we can not access the value of these variables with only 1 or 2 lines of code.

Dr. M. Raja Roy, Associate Professor, ANITS

www.mrrtechnical.co.in

Array

Another way to do this is array. By using array, we can access the elements easily. Only few lines of code is required to access the elements of array.

Array

Advantage of C Array

- 1) Code Optimization:** Less code to the access the data.
- 2) Easy to traverse data:** By using the for loop, we can retrieve the elements of an array easily.
- 3) Easy to sort data:** To sort the elements of array, we need a few lines of code only.
- 4) Random Access:** We can access any element randomly using the array.

Array

Disadvantage

Fixed Size: Whatever size, we define at the time of declaration of array, we can't exceed the limit. So, it doesn't grow the size dynamically like LinkedList which we will learn later.

Declaration of C Array

```
data_type array_name[array_size];
```

Ex:

```
Int marks[5];
```

Dr. M. Raja Roy, Associate Professor, ANITS

www.mrrtechnical.co.in

Array

Initialization of Array

A simple way to initialize array is by index. Notice that array index starts from 0 and ends with [SIZE - 1].

```
marks[0]=80;//initialization of array
```

```
marks[1]=60;
```

```
marks[2]=70;
```

```
marks[3]=85;
```

```
marks[4]=75;
```

80	60	70	85	75
marks[0]	marks[1]	marks[2]	marks[3]	marks[4]

Initialization of Array

Dr. M. Raja Roy, Associate Professor, ANITS

www.mrrtechnical.co.in

Array

```
#include<stdio.h>
void main()
{
    int i;
    int marks[5]; //Declaration of array
    //initialization of array
    marks[0]=80;
    marks[1]=70;
    marks[2]=60;
    marks[3]=50;
    marks[4]=40;
    //traversal of array
    for(i=0;i<5;i++)
    {
        printf("\n Marks[%d] : %d",i,marks[i]);
    }
}
```

Out Put :

```
Marks[0] : 80
Marks[1] : 70
Marks[2] : 60
Marks[3] : 50
Marks[4] : 40
```

Dr. M. Raja Roy, Associate Professor, ANITS

www.mrrtechnical.co.in

Array

Array: Declaration with Initialization

Syntax to initialize the an array at the time of declaration.

```
int marks[5]={20,30,40,50,60};
```

Dr. M. Raja Roy, Associate Professor, ANITS

www.mrrtechnical.co.in

Array

```
#include<stdio.h>
void main()
{
    int i;
    //Declaration with Initialization
    int marks[5]={80,70,60,50,40};

    //traversal of array
    for(i=0;i<5;i++)
    {
        printf("\n Marks[%d] : %d",i,marks[i]);
    }
}
```

Out Put :

```
Marks[0] : 80
Marks[1] : 70
Marks[2] : 60
Marks[3] : 50
Marks[4] : 40
```

Dr. M. Raja Roy, Associate Professor, ANITS

www.mrrtechnical.co.in

Array

Array: Declaration with Initialization

Syntax to initialize the an array at the time of declaration.

In such case, there is no requirement to define size. So it can also be written as the following code.

```
int marks[]={20,30,40,50,60};
```

Dr. M. Raja Roy, Associate Professor, ANITS

www.mrrtechnical.co.in

Array

```
#include<stdio.h>
void main()
{
    int i;
    //Declaration with Initialization
    int marks[]={80,70,60,50,40};

    //traversal of array
    for(i=0;i<5;i++)
    {
        printf("\n Marks[%d] : %d",i,marks[i]);
    }
}
```

Out Put :

```
Marks[0] : 80
Marks[1] : 70
Marks[2] : 60
Marks[3] : 50
Marks[4] : 40
```

Dr. M. Raja Roy, Associate Professor, ANITS

www.mrrtechnical.co.in

Two Dimensional Array

The two dimensional array in C language is represented in the form of rows and columns, also known as matrix. It is also known as array of arrays or list of arrays.

Declaration of two dimensional Array

Syntax :

```
data_type array_name[size1][size2];
```

Ex: **int** twodimen[4][3];

Here, 4 is the row number and 3 is the column number.

Dr. M. Raja Roy, Associate Professor, ANITS

www.mrrtechnical.co.in

Two Dimensional Array

Initialization of 2D Array

A way to initialize the two dimensional array at the time of declaration is given below.

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

2D Array

```
#include <stdio.h>
#include <conio.h>
void main()
{
    int i=0,j=0;
    int arr[4][3]={{1,2,3},{2,3,4},{3,4,5},{4,5,6}};

    //traversing 2D array
    for(i=0;i<4;i++)
    {
        for(j=0;j<3;j++)
        {
            printf("arr[%d] [%d] = %d \n",i,j,arr[i][j]);
        }
    }
    getch();
}
```

Out Put :

```
arr[0] [0] = 1
arr[0] [1] = 2
arr[0] [2] = 3
arr[1] [0] = 2
arr[1] [1] = 3
arr[1] [2] = 4
arr[2] [0] = 3
arr[2] [1] = 4
arr[2] [2] = 5
arr[3] [0] = 4
arr[3] [1] = 5
arr[3] [2] = 6
```


2D Array : Dynamic input

```
#include <stdio.h>
void main ()
{
    int arr[3][3],i,j;
    for (i=0;i<3;i++)
    {
        for (j=0;j<3;j++)
        {
            printf("Enter a[%d][%d]: ",i,j);
            scanf("%d",&arr[i][j]);
        }
    }
    printf("\n printing the elements ....\n");
    for(i=0;i<3;i++)
    {
        printf("\n");
        for (j=0;j<3;j++)
        {
            printf("%d\t",arr[i][j]);
        }
    }
}
```

Out Put :

```
Enter a[0][0]: 56
Enter a[0][1]: 10
Enter a[0][2]: 30
Enter a[1][0]: 34
Enter a[1][1]: 21
Enter a[1][2]: 34
```

```
Enter a[2][0]: 45
Enter a[2][1]: 56
Enter a[2][2]: 78
```

printing the elements

```
56  10  30
34  21  34
45  56  78
```

Dr. M. Raja Roy, Associate Professor, ANITS

www.mrrtechnical.co.in

Three Dimensional Array

The three dimensional array in C language is represented multiple two dimensional arrays in the form of rows and columns.

Declaration of three dimensional Array

Syntax :

```
data_type array_name[size1][size2][size3];
```

Ex: `int arex[2][4][3];`

Here, 2 sets of matrices with 4 rows and 3 columns will be displayed.

Dr. M. Raja Roy, Associate Professor, ANITS

www.mrrtechnical.co.in

3D Array

```
#include <stdio.h>
int main()
{
    int arex[2][4][3],i,j,k;
    printf("Enter Elements into 3D Array : ");
    for(i=0;i<2;i++)
    {
        for(j=0;j<4;j++)
        {
            for(k=0;k<3;k++)
            {
                printf("\narex[%d][%d][%d] : ",i,j,k);
                scanf("%d",&arex[i][j][k]);
            }
        }
    }

    //Code to Display 3D Array
    printf("\n3D Array");
}
```

```
for(i=0;i<2;i++)
{
    printf("\n2D Array-%d",i+1);
    for(j=0;j<4;j++)
    {
        printf("\n");
        for(k=0;k<3;k++)
        {
            printf("\t%d\t",arex[i][j][k]);
        }
    }
    return 0;
}
```

Dr. M. Raja Roy, Associate Professor, ANITS

www.mrrtechnical.co.in

3D Array

Out Put :

Enter Elements into 3D Array :

```
arex[0][0][0] : 1
arex[0][0][1] : 2
arex[0][0][2] : 3
arex[0][1][0] : 4
arex[0][1][1] : 5
arex[0][1][2] : 6
arex[0][2][0] : 7
arex[0][2][1] : 8
arex[0][2][2] : 9
arex[0][3][0] : 10
arex[0][3][1] : 11
arex[0][3][2] : 12
arex[1][0][0] : 13
arex[1][0][1] : 14
arex[1][0][2] : 15
arex[1][1][0] : 16
arex[1][1][1] : 17
arex[1][1][2] : 18
arex[1][2][0] : 19
arex[1][2][1] : 20
arex[1][2][2] : 21
arex[1][3][0] : 22
arex[1][3][1] : 23
arex[1][3][2] : 24
```

3D Array

2D Array-1

1	2	3
4	5	6
7	8	9
10	11	12

2D Array-2

13	14	15
16	17	18
19	20	21
22	23	24

Dr. M. Raja Roy, Associate Professor, ANITS

www.mrrtechnical.co.in

Array operations : Inserting a value

```
// Online C compiler to run C program online
#include <stdio.h>
int main() {
    int ar[7]={10,20,30,40,50},i,pos,n;
    printf("\nPrint Array");
    for(i=0;i<5;i++)
    {
        printf("\n%d",ar[i]);
    }
    printf("\nEnter position to inset:");
    scanf("%d",&pos);
    printf("\nEnter value:");
    scanf("%d",&n);

    //Code to shift
    for(i=5;pos<=i;i--)
    {
        ar[i+1]=ar[i];
        //printf("%d",i);
    }
}
```

```
ar[pos]=n;
printf("\nPrintNew Array");

for(i=0;i<6;i++)
{
    printf("\n%d",ar[i]);
}
return 1000;
}
```

```
Out Put :
Print Array
10
20
30
40
Enter position to inset:2
Enter value:100
PrintNew Array
10
20
100
40
50
```

Dr. M. Raja Roy, Associate Professor, ANITS

www.mrrtechnical.co.in

Array operations : Delete a value

```
// Online C compiler to run C program online
#include <stdio.h>

int main() {

    int ar[7]={10,20,30,40,50},i,pos,n;

    printf("\nPrint Array");
    for(i=0;i<4;i++)
    {
        printf("\n%d",ar[i]);
    }

    printf("\nEnter position to delete:");
    scanf("%d",&pos);

    //Code to shift
    for(i=pos;i<5;i++)
    {
        ar[i]=ar[i+1];
    }
}
```

```
printf("\nPrintNew Array");
for(i=0;i<4;i++)
{
    printf("\n%d",ar[i]);
}
return 1000;
}
```

```
Out Put :
Print Array
10
20
30
40
Enter position to delete:1

PrintNew Array
10
30
40
```

Dr. M. Raja Roy, Associate Professor, ANITS

www.mrrtechnical.co.in