Introduction to arrays

- Array is a collection of similar datatype
- Declaring array:

datatype arrayname[index];

Example : int list[20];

char num[10];

float arr[12];

Num[0] – represents the first element in array

Num[9] - represents the last element in array

Introduction to arrays

- Contiguous memory allocation is used to store array elements
- After creating an array its size cannot be changed.
- If array size is arr[10] we cannot increase or decrease its size.
- Inserting an element at arr[11] is not possible

Initializing arrays (one dimensional array)

- Declaring, creating , initializing array

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

    studentRegno[0]=2100;
    studentRegno[1]=2101;
    studentRegno[2]=2102;
    studentRegno[3]=2103;
    studentRegno[4]=2104;
    studentRegno[5]=2106;
```

Types of arrays

There are 2 types of arrays.

They are,

- One dimensional array
- Multi dimensional array
  - Two dimensional array
  - Three dimensional array

One Dimensional array

Example

```
#include <stdio.h>
 int main( )
{
 int val[7] = { 11, 22, 33, 44, 55, 66, 77 } ;
  for ( int i = 0 ; i < 7 ; i++ )
 {
printf("val[%d]: value is %d and address is %d\n", i, val[i]);
}
return 0;
}
```

One dimensional array - example

```
#include<stdio.h>
 int main()
{
    int i;
    int arr[5] = {10,20,30,40,50};

      // declaring and Initializing array in C
      //To initialize all array elements to 0, use int arr[5]={0};
      /* Above array can be initialized as below also
      arr[0] = 10;
      arr[1] = 20;
      arr[2] = 30;
      arr[3] = 40;
      arr[4] = 50; */
     for (i=0;i<5;i++)
    {
            printf("value of arr[%d] is %d \n", i, arr[i]);
     }
```

Multidimensional array:

- An array of arrays is called as multi dimensional array
- Array created with more than one dimension(size)

Syntax:

```
datatype arrayname[rowSize][columnsize];
```

Example: `int matrix_A[2][3];`

Initialization of two dimensional array:

```
int matrix_A [2][3] = { {1, 2, 3},{4, 5, 6} } ;  (or)
int matrix_A [2][3] = { {1, 2, 3},
                        {4, 5, 6} } ;
```

Two dimensional array - Example

```
#include<stdio.h>
 int main()
int disp[2][3];
 int i, j;
for(i=0; i<2; i++)
{
 for(j=0;j<3;j++)
{
 printf("Enter value for disp[%d][%d]:", i, j);
 scanf("%d", &disp[i][j]);
}
 }
  printf("Two Dimensional array elements:\n");
 for(i=0; i<2; i++)
{ for(j=0;j<3;j++)
 {
printf("%d ", disp[i][j]);
if(j==2)
{
 printf("\n");
 }
 }
} return 0;
 }
```

Applications of array

- Arrays are used to Store List of values
- Arrays are used to Perform Matrix Operations
- Arrays are used to implement Search Algorithms
- Arrays are used to implement Sorting Algorithms
- Arrays are used to implement Datastructures

THANK YOU