

Unit-6

❖ Introduction of Array (one dimensional)

- Array Terminology
- A characteristics of an array
- Array Declarations
- Array initialization
- Accessing an array
- Storing value in an array(Bubble Sort)

❖ What is Array?

- An array is a collection of variable of same data type that share a same name.
- For ex: `int a[10];`
- Here a is an array of 10 integer elements.
- Array can be divided into two type.
 - One dimensional array.
 - two dimensional array
 - multi dimensional array.

❖ What are the characteristics of arrays in C?

- An array holds elements that have the same data type
- Array elements are stored in subsequent memory locations
- Two-dimensional array elements are stored row by row in subsequent memory locations.
- Array name represents the address of the starting element
- Array size should be mentioned in the declaration. Array size must be a constant expression and not a variable
- While declaring the 2D array, the number of columns should be specified and it's a mandatory. whereas for number of rows there is no such rule.

❖ Declaration of C Array

- We can declare an array in the c language in the following way.
 - `data_type array_name[array_size];`
- Now, let us see the example to declare array.
 - `int marks[5];`
- `int` is the *data_type*
- `marks` is the *array_name*
- And 5 is the *array_size*.

❖ What is one-dimensional array? Explain with example.

- A list of items can be given one variable name using only one subscript. Such a variable is called a single subscripted variable or one dimensional array.
- Subscript variable consist of array name and the position of the element in the array.
- **Syntax:**
 - `Data type arrayname[size];`
- Data type: Data types define the type of the array element, like `int`, `float`, `char`, `long int` etc.
- Array name: array name is the name of variable which represent array.
- Size: which is represents within `[]` symbol represent the size of the array.

- For **example**: `int a[10];`
 - Here `int` is a data type.
 - `a` is array variable name.
 - `10` is the size of the array.
- In 'c' language array index or subscript start from 0, so in the above example 10 different integer variable as, `a[0]`, `a[1]`, `a[2]`, `a[3]`, `a[4]`, `a[5]`, `a[6]`, `a[7]`, `a[8]` and `a[9]`.
- Where `a[0]` is the first element of the array, while `a[9]` is the last element of the array.

<code>a[0]</code>	<code>a[1]</code>	<code>a[2]</code>	<code>a[3]</code>	<code>a[4]</code>	<code>a[5]</code>	<code>a[6]</code>	<code>a[7]</code>	<code>a[8]</code>	<code>a[9]</code>
12	21	45	2	56	43	67	3	8	1
0	1	2	3	4	5	6	7	8	

- From the above figure we have 10 different variable, `a[0]..a[9]`. the index value goes from 0 to 9.
- Example:-

```
#include<stdio.h>
#include<conio.h>
void main()
{
    int a[5]; /* a is an array of 5 integers */
    int i,j;
    for(i=0;i<5;i++)
    {
        a[i]= i + 1;
    }
    /* output each array element's value */
    for(j=0;j<5;j++)
    {
        printf("a[%d]= %d\n", j, a[j] );
    }
    getch();
}
```

❖ Explain initialization of one dimensional array.

- When the elements of the array are fixed and there is no need to take the value from the keyboard then we can initialize the array at the time of the declaration.
- Format:
 - `Type array_name[size]={list of value};`
- Ex: `int A[4]={1,2,3,4};`
- The value in the least is separated by comma.
- Initialize array A as follow.
- `A[0]=1,A[1]=2,A[2]=3,A[3]=4.`
- For non-ANSI compiler above initialize is done as follows.
- `Static int A[4]={1,2,3,4};`
- If the number of the value are less then remaining element then the remaining element are initialize to zero.
- For ex: `int A[5]={1,2,3};`
- Initialize array A as follow.

- A[0]=1,A[1]=2,A[2]=3,A[3]=0,A[4]=0.
- If the size of the array is not given in the statement then it can be written as follows.
- Float s[]={1.1,1.2,1.3,1.4};
- **Example:-**

```
#include <stdio.h>
#include <conio.h>
void main()
{
    int i;
    int marks[5]={20,30,40,50,60}; //declaration and initialization of array
    clrscr();
    for(i=0;i<5;i++)
    {
        printf("%d \n",marks[i]);
    }
    getch();
}
```

❖ Accessing Array Elements

- An element is accessed by indexing the array name.
- This is done by placing the index of the element within square brackets after the name of the array.
- For example:- int b=a[9];
- The above statement will take the 10th element from the array and assign the value to salary variable.
- The following example Shows how to use all the three above mentioned concepts viz. declaration, assignment, and accessing arrays –
- Example:-

```
#include <stdio.h>
#include <conio.h>
void main()
{
    int a[3]={20,30,40}; //declaration and initialization of array
    clrscr();
    printf("%d \n",a[1]);
    getch();
}
```

❖ Explain Advantage and Disadvantage of C Array

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

- **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 Linked List which we will learn later.

❖ C Program to Sort N Numbers in Ascending Order using Bubble Sort

```
#include<stdio.h>
#include<conio.h>
void main()
{
    int a[10],i ,j ,n ,temp;
    printf("Enter the value of num \n");
    scanf("%d", &n);
    printf("Enter the elements one by one \n");
    for(i=0 ;i<n ;i++)
    {
        scanf("%d", &a[i]);
    }
    for (i = 0; i < n; i++)
    {
        for (j = 0; j < (n - i - 1); j++)
        {
            if (a[j] > a[j + 1])
            {
                temp = a[j];
                a[j] = a[j + 1];
                a[j + 1] = temp;
            }
        }
    }
    printf("Sorted array is...\n");
    for (i = 0; i < n; i++)
    {
        printf("%d\n", a[i]);
    }
}
```