Chapter: 7
ARRAYS & STRINGS

Department of Electronics and Computer Engineering

Er. Bigyan Karki

Er.Bikash Acharya

Er. Pradip Khanal

ADVANCED COLLEGE OF ENGINEERING & MANAGEMENT

ESTD. 2000

Affiliated to Tribhuvan University (T.U.)

ARRAY

- An array is a group of related data items that share a common name.

  i.e. a data structure as a single entity

- Simple variable is a single memory location with unique name and a type whereas an array is a collection of adjacent memory locations that have one collection name and type.

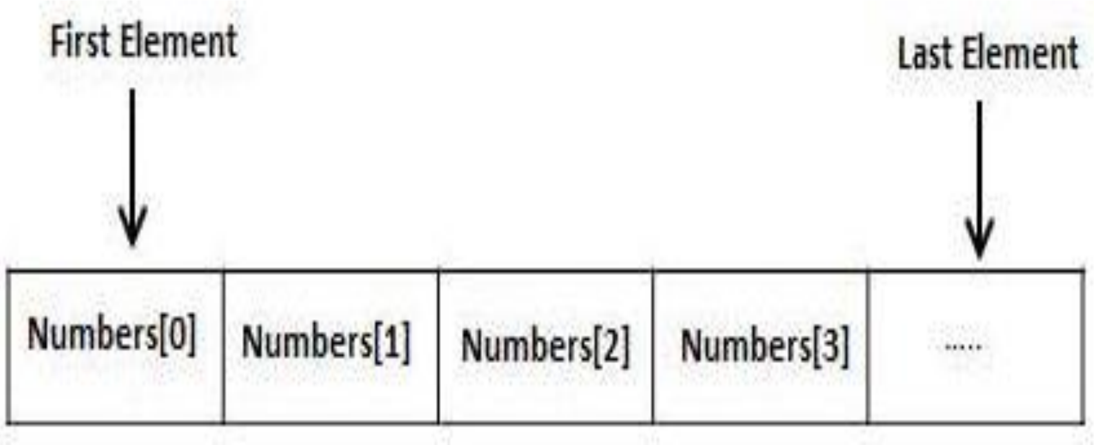
- The individual values are called **“elements”** in an array and

  length of an array = number of elements.

- Each element is identified by its position number or **“index” or “Subscript”** in an array. [An index always begins with value **0**].

ARRAY

- All arrays consist of contiguous memory locations.
- The lowest address corresponds to the first element and the highest address to the last element.



Motivation

- If we have to store information of 100 students. 100 students.
- Without array : we use

  int sr1; int sr2; int sr3; ....; int sr100; (Inefficient Task)
- With array : We use int sr[100];
- can store 100 students information. (Efficient Task) sr[0],sr[1],...................................,sr[99] ,

  here 0-99 are index.

Advantages of Arrays

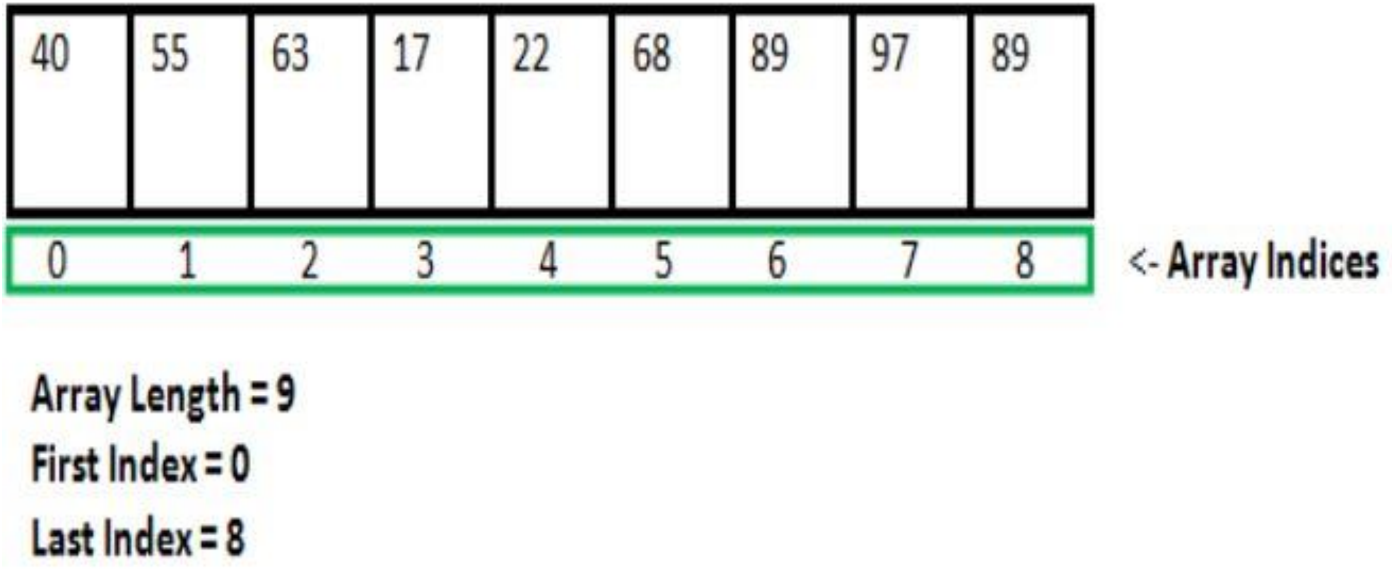
- Arrays can store a large number of values with single name.
- Value stored in arrays can be accessed/processed easily and quickly and can be sorted and conduct searching process easily.
- Array can be used for matrix computations as well.

Types of Array

- One-Dimensional Array
- Two-Dimensional Array
- Multi-Dimensional Array

One-Dimensional Array

- One-Dimensional Array is also called as linear array i.e 1-D array.
- It stores data in a single row or column.



40	55	63	17	22	68	89	97	89	
0	1	2	3	4	5	6	7	8	<- Array Indices

Array Length = 9

First Index = 0

Last Index = 8

- **Declaration of 1-D Array**

  Syntax:

  storage_class data_type array_name[size]

  int num[5];

- **Initialization of 1-D Array**
  - The process of assigning values to array elements at the time of array declaration is called array initialization.

  Syntax:

  storage_class data_type array_name[size] = [value1,value2,………]

- **Declaration of 1-D Array**

  Syntax:

  storage_class data_type array_name[size]

  int num[5];

- **Initialization of 1-D Array**
  - The process of assigning values to array elements at the time of array declaration is called array initialization.

  Syntax:

  storage_class data_type array_name[size] = [value1,value2,………]

- **The array initialization can be of following types:**

  i. int a[5] = {1,2,3,4,5};

  here, array ‘a’ has 5 elements and values are assigned as (a[0]=1, a[1]=2, a[2]=3, a[3]=4, a[4]=5)

  ii. int b[ ] ={ 2,5,7};

  here, the size of array is automatically set by compiler ,according to number of values given or number of elements.

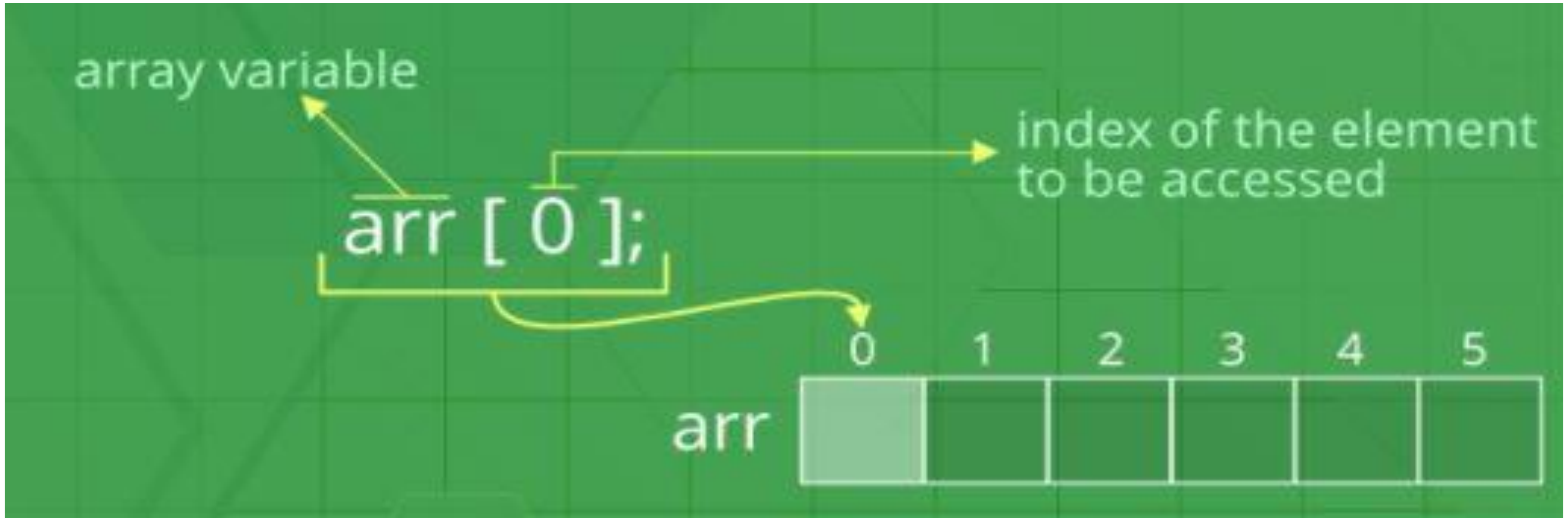
  Also, identical to: **int b[3] ={2,5,7};**

  iii. int c[8] = {45,56,78,44,23};

  here,array size is set to 8 but only 5 elements, so c[5], c[6], c[7] is **‘0’ zero** in this example.

- **Accessing Array Elements**

- Array elements are accessed by using an integer index.



- All elements of an array can neither be set at once nor one array may be assigned to others.
- For Example: int a[5], b[5];

  a=0;   // *Wrong //*

  b=a; // *Wrong //*

  if(a<b>

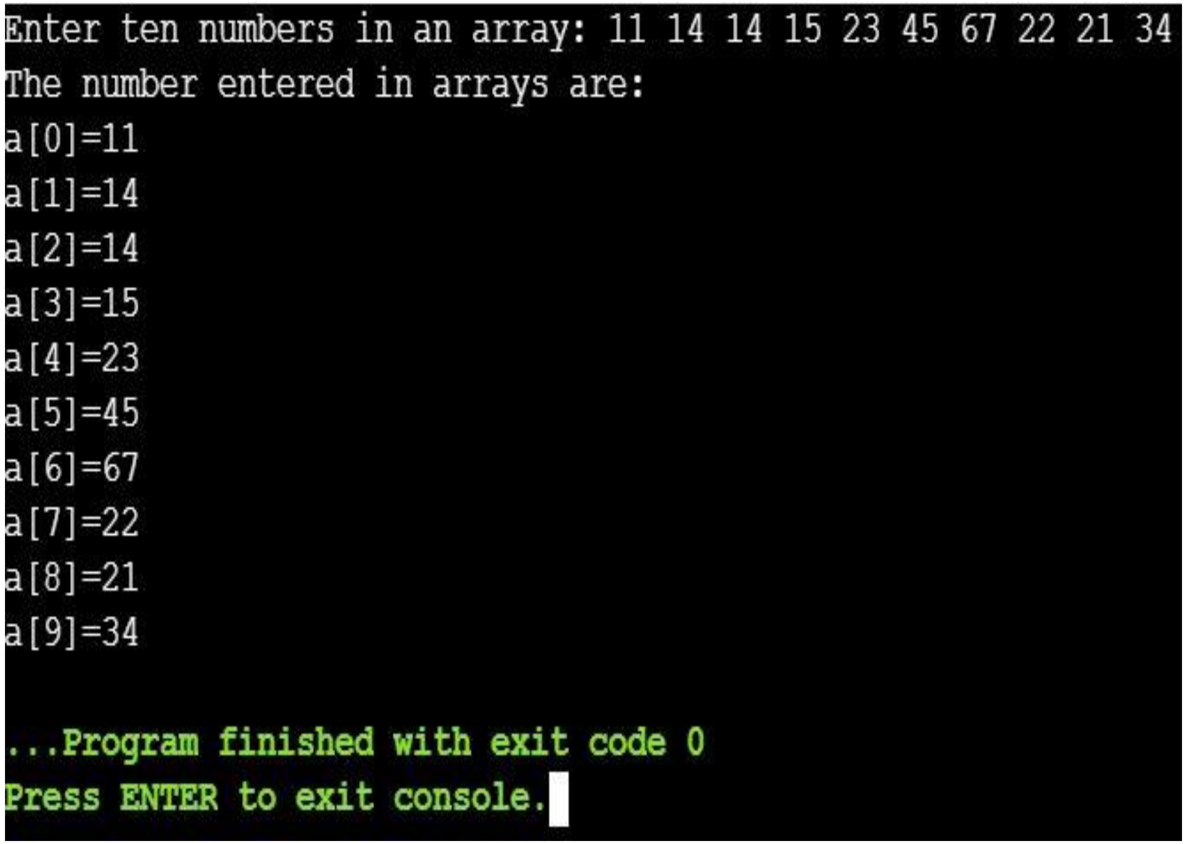
  {…. // *Wrong //….}*

EXAMPLE:

- WAP that reads 10 integers from user, stores in an array and displays the entered numbers

```c
#include<stdio.h>
#include<conio.h>
int main()
{
 int a[10],i;
 printf("Enter ten numbers in an array: ");
 for(i=0;i<10;i++){
 scanf("%d",&a[i]);  //Array Input//
 }
 printf("The number entered in arrays are:");
 for(i=0;i<10;i++){
 printf("\na[%d]=%d",i,a[i]);
}
 getch();
   return 0;
}
```

OUTPUT



```
Enter ten numbers in an array: 11 14 14 15 23 45 67 22 21 34
The number entered in arrays are:
a[0]=11
a[1]=14
a[2]=14
a[3]=15
a[4]=23
a[5]=45
a[6]=67
a[7]=22
a[8]=21
a[9]=34


...Program finished with exit code 0
Press ENTER to exit console.
```