Unit : 3

Array

Arrays

- Arrays a kind of data structure that can store a fixed-size sequential collection of elements of the same type. An array is used to store a collection of data, but it is often more useful to think of an array as a collection of variables of the same type.

- Instead of declaring individual variables, such as number0, number1, ..., and number99, you declare one array variable such as numbers and use numbers[0], numbers[1], and ..., numbers[99] to represent individual variables. A specific element in an array is accessed by an index.
- All arrays consist of contiguous memory locations. The lowest address corresponds to the first element and the highest address to the last element.

First					Last
Number[0]	Number[1]	Number[2]	Number[3]	Number[4]	Number[5]

Declaring Arrays

- To declare an array in C, a programmer specifies the type of the elements and the number of elements required by an array as follows

- Type array_name [Size]

- This is called a *single-dimensional* array. The **arraySize** must be an integer constant greater than zero and **type** can be any valid C data type. For example, to declare a 10-element array called **balance** of type double, use this statement.

- double balance[10];

- Here *balance* is a variable array which is sufficient to hold up to 10 double numbers.

Initializing Arrays

- You can initialize an array in C either one by one or using a single statement as follows.
- double balance[5] = {1000.0, 2.0, 3.4, 7.0, 50.0};

- The number of values between braces { } cannot be larger than the number of elements that we declare for the array

- If you omit the size of the array, an array just big enough to hold the initialization is created. Therefore, if you write – between square brackets [ ].

- double balance[] = {1000.0, 2.0, 3.4, 7.0, 50.0};

- You will create exactly the same array as you did in the previous example. Following is an example to assign a single element of the array.

- balance[4] = 50.0;

- The above statement assigns the 5th element in the array with a value of 50.0. All arrays have 0 as the index of their first element which is also called the base index and the last index of an array will be total size of the array minus 1. the array index was started from zero(0).

Two Dimension Array

- The simplest form of multidimensional array is the two-dimensional array. A two-dimensional array is, in essence, a list of one-dimensional arrays. To declare a two-dimensional integer array of size [x][y], you would write something as follows,

- Type name_array [x][y];

Ex. int balance[3][3]

Balance	0	1	2
0			
1		500	
2			

```
Balance [1][1] = 500;
```

Three dimension array

- The three dimension array collection of rows and columns.

- To declare three dimension array use following syntax

```
Type  array_name [size][rows][columns]

int ip[2][2][2]
```

* Above statement creates array of 8 elements

int bal[3][3][3]

	0					1		
	0	1	2			0	1	2
0	100				0	200		
1					1			
2					2			

Bal[0][0][0] = 100

Bal[1][0][0] = 200

Assign Values to array

- You can also assign values to array when declaring array.

- Ex
  - int marks [5] = {45,67,54,24,78};

  - So marks[0] becomes 45, marks[1] becomes 67 and so on

Character array

- Also you can create array of characters, its known as string.

- Ex.
  - Char name[10] = “rajkot”;

  String must be specified in Double quote only(“”),

  In the above example index of **r** is 0, **a** is 1 and so on.

Programs from this Unit

- Size of array
- Store values in array
- Sum of array
- Maximum out of array
- Minimum out of array
- sort array in ascending order
- Sort array in descending order.
- Sum of two array
- Print sum of array matrix.

- Print two dimension Array.
- Print three dimension array.
- Initialize array.
- Character array.
- Find maximum out of 2d array,
- Make sum of 2d array and store in another array