

Chapter 2

Arrays

An **array** is a collection of elements of the same type placed in contiguous memory locations. Arrays are used to store a set of values of the same type under a single variable name. Each element in an array can be accessed using its position in the list called index number or subscript.

The syntax for declaring an array in C++ is as follows.

```
data_type array_name[size];
```

Eg: `int num[10];`

This statement declares an array named `num` that can store 10 integer numbers.

Memory size of an array

For a single dimensional array, the total size allocated can be found using the following formula:

$$\text{total_bytes} = \text{sizeof}(\text{array_type}) \times \text{size_of_array}$$

The array in the above example requires $4 \times 10 = 40$ Bytes

Array initialisation

Array elements can be initialised in their declaration statements, as shown in the following example:

```
int score[5] = {98, 87, 92, 79, 85};
```

In this case, `score[0]` contains 98, `score[1]` contains 87 and `score[4]` contains 85.

Accessing array elements

Elements of an array are easily accessed using a `for` loop. For example, the elements in the above array can be displayed using the following loop:

```
for (i=0; i<5; i++)
    cout<<score[i];
```

String handling using arrays

A character array can be used to store a string, since it is a sequence of characters. The array `char my_name[10];` can store a string of 9 characters. One location will be used to store '\0' (null character) as string terminator.

A string can be input using the statement:

```
cin >> my_name;
```

This statement can store a string without any white space (that is, only one word). If we want store strings containing white spaces (strings having more than one word) we can use `gets()` function, `getline()` function or `get()` function.

`cin.getline(my_name, 10);` can accept a string containing white spaces. The functions `getline()` and `get()` are stream functions to input string and `gets()` is a console function.

Similarly to display string data we can use `puts()` function and `write()` function.

Questions from Previous Years' Question Papers

1. Write a C++ program to accept a string and count the number of words and vowels in that string. (5) (March 2016)
2. Write a C++ program to accept N integer numbers and find the sum and average of even numbers. Use array to store numbers. (5) (March 2016)
3. How memory is allocated for a float array? (2) (March 2016)
4. How can we initialize an integer array? Give an example. (2) (March 2016)
5. Write a C++ program to accept a string and find the length of the string without using built-in function. Use a character array to store the string. (5) (SAY 2016)
6. Write a program to input 'N' numbers into an integer array and find the largest number in the array. (5) (SAY 2016)
7. Define an array. Give an example of an integer array declaration. (2) (SAY 2016)
8. Consider the following C++ code:

```
char text[20];  
cin>>text;  
cout<<text;
```

If the input string is "Computer Programming", what will be the output? Justify your answer. (2) (SAY 2016)

9. Write C++ initialization statement to initialize an integer array name 'MARK' with the values 70, 80, 85, 90. (1) (March 2017)
10. Write a C++ program to input 10 numbers into an array and find the sum of the numbers which are exact multiples of 5. (3) (March 2017)
11. Explain two stream functions for input operation with example. (2) (March 2017)
12. Explain how allocation of string takes place in memory. (2) (SAY 2017)
13. Explain gets() and puts() functions. (2) (SAY 2017)
14. Write a C++ program to enter 10 numbers into an array and find the second largest element. (5) (SAY 2017)
15. Write a C++ program to convert all lowercase alphabets stored in a string to uppercase. (5) (SAY 2017)