# Lab 8

# Array

C++ provides a data structure, **the array**, which stores 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.

**Array Declaration**

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

type arrayName [ arraySize ];

**Array Initialization**

You can initialize C++ array elements either one by one or using a single statement as follows –

double balance[5] = {1000.0, 2.0, 3.4, 17.0, 50.0};

The number of values between braces { } can not be larger than the number of elements that we declare for the array between square brackets [ ]. Following is an example to assign a single element of the array −

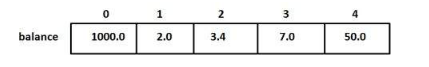
If you omit the size of the array, an array just big enough to hold the initialization is created. Therefore, if you write −

double balance[] = {1000.0, 2.0, 3.4, 17.0, 50.0};

You will create exactly the same array as you did in the previous example.

balance[4] = 50.0;

The above statement assigns element number 5th in the array a value of 50.0. Array with 4th index will be 5th, i.e., last element because all arrays have 0 as the index of their first element which is also called base index. Following is the pictorial representation of the same array we discussed above −



**Accessing an Array**

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 –

double salary = balance[9];

The above statement will take 10th element from the array and assign the value to salary variable. Following is an example, which will use all the above-mentioned three concepts viz. declaration, assignment and accessing arrays −

**Program 1**

#include <iostream>

using namespace std;

#include <iomanip>

using std::setw;

int main () {

int n[ 10 ]; // n is an array of 10 integers

// initialize elements of array n to 0

for ( int i = 0; i < 10; i++ ) {

n[ i ] = i + 100; // set element at location i to i + 100

}

cout << "Element" << setw( 13 ) << "Value" << endl;

// output each array element's value

for ( int j = 0; j < 10; j++ ) {

cout << setw( 7 )<< j << setw( 13 ) << n[ j ] << endl;

}

**Output:**

Element Value

0 100

1 101

2 102

3 103

4 104

5 105

6 106

7 107

8 108

9 109

**Program 2**

#include <iostream>

#include<conio.h>

using namespace std;

#define ARRAY\_SIZE 5

int main()

{

int array[ARRAY\_SIZE];

int i;

float total = 0.0, average;

cout<<"Sum of Array In C++ Example Program\n";

for (i = 0; i < ARRAY\_SIZE; i++)

{

cout<<"Enter the Number : "<< (i+1) <<" : ";

cin>>array[i];

}

/\* Sum of the Array Using for Loop \*/

for (i = 0; i < ARRAY\_SIZE; i++)

{

total = total + array[i] ;

}

average = total / ARRAY\_SIZE;

cout<<"\n Sum of all Numbers = "<< total<<"\n";

cout<<"\n Average of all input numbers = "<< average<<"\n";

getch();

}

**Program 3**

*/ Header Files*

**#include <iostream>**

**#include<conio.h>**

using namespace std;

int **main**()

{

int i;

*// declaring and Initialising array in C*

int value[6] = {5,10,15,20,25,30};

cout<<"Single Dimensional Array In C++ Example Program\n";

for (i=0;i<6;i++)

{

*// Accessing each variable using for loop*

cout<<"Position : "<<i<<" , Value : "<< value[i]<<" \n";

}

*// Wait For Output Screen*

getch();

*//Main Function return Statement*

return 0;

}

**Output;**

Position : [0] , Value : 5

Position : [1] , Value : 10

Position : [2] , Value : 15

Position : [3] , Value : 20

Position : [4] , Value : 25

Position : [5] , Value : 30

*Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Marks Obtained: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Instructor’s Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*