C++ Arrays

An array is a collection of data that holds fixed number of values of same type. For example:

int age[100];

Here, the age array can hold maximum of 100 elements of integer type.

The size and type of arrays cannot be changed after its declaration.

## How to declare an array in C++?

dataType arrayName[arraySize];

For example,

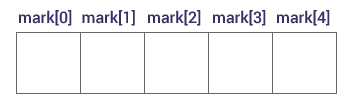
float mark[5];

Here, we declared an array, mark, of floating-point type and size 5. Meaning, it can hold 5 floating-point values.

## Elements of an Array and How to access them?

You can access elements of an array by using indices.

Suppose you declared an array mark as above. The first element is mark[0], second element is mark[1] and so on.



### Few key notes:

* Arrays have 0 as the first index not 1. In this example, mark[0] is the first element.
* If the size of an array is n, to access the last element, (n-1) index is used. In this example, mark[4] is the last element.
* Suppose the starting address of mark[0] is 2120d. Then, the next address, a[1], will be 2124d, address of a[2] will be 2128d and so on. It's because the size of float is 4 bytes.

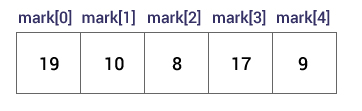
## How to initialize an array in C++ programming?

It's possible to initialize an array during declaration. For example,

int mark[5] = {19, 10, 8, 17, 9};

Another method to initialize array during declaration:

int mark[] = {19, 10, 8, 17, 9};



Here,

mark[0] is equal to 19

mark[1] is equal to 10

mark[2] is equal to 8

mark[3] is equal to 17

mark[4] is equal to 9

## How to insert and print array elements?

int mark[5] = {19, 10, 8, 17, 9}

// change 4th element to 9

mark[3] = 9;

// take input from the user and insert in third element

cin >> mark[2];

// take input from the user and insert in (i+1)th element

cin >> mark[i];

// print first element of the array

cout << mark[0];

// print ith element of the array

cout >> mark[i-1];

## Example: C++ Array

**C++ program to store and calculate the sum of 5 numbers entered by the user using arrays.**

#include <iostream>

using namespace std;

int main()

{

int numbers[5], sum = 0;

cout << "Enter 5 numbers: ";

// Storing 5 number entered by user in an array

// Finding the sum of numbers entered

for (int i = 0; i < 5; ++i)

{

cin >> numbers[i];

sum += numbers[i];

}

cout << "Sum = " << sum << endl;

return 0;

}

**Output**

Enter 5 numbers: 3

4

5

4

2

Sum = 18

## Things to remember when working with arrays in C++

Suppose you declared an array of 10 elements. Let's say,

int testArray[10];

You can use the array members from testArray[0] to testArray[9].

If you try to access array elements outside of its bound, let's say testArray[14], the compiler may not show any error. However, this may cause unexpected output (undefined behavior).