7.

Date

Arrays

In C++, an array is a variables that can store multiple values of the same type. For example,

Suppose a class has 27 students, & we need to store & grades of all of them. Instead of creating 27 seprate variables, we can simply create an array.

```
double grade[27];
```

Here, grade is an array that can hold a maximum of 27 elements of double type.

In C++, the size & type of array cannot be changed after its decleration.

C++ Array Decleration :

```
dataType arrayName[arraySize];
```

For example,

```
int x[6]
```

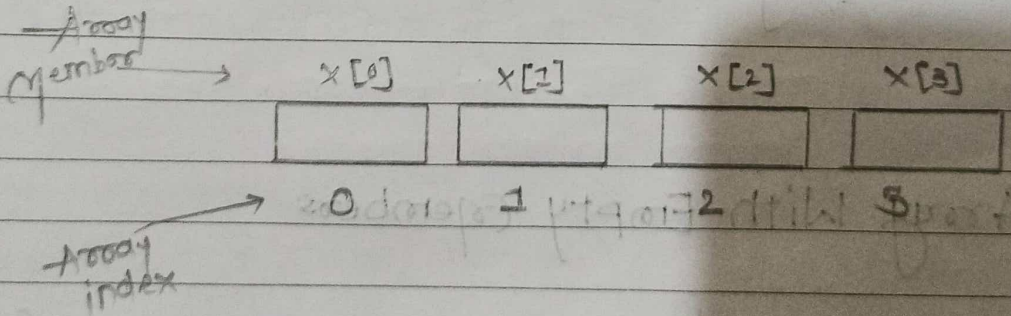
Here,

- int - type of element to be stored.
- x - Named the array.
- 6 - Sized of the array.

Access Element in C++ Array.

In C++, each element is an array is associated with a number. The number is known as an array index. We can access element of an array by using those indices.

```
// Syntax to access array element
array[index];
```



few things to Remember

- The array indices start with 0. Meaning x[0] is the first stored at index 0.

Saathi

Date ___ / ___ / _____

- If the size of an array is n, the last element is stored at index (n-1). In this example, x[5] is last element.

C++ Array Initilization

In C++, it's possible to initilize an array during decleration. For example.

```
// declare & initilize an array
int x[6] = {19, 10, 8, 17, 9, 15};
```

Here, we have not mentioned the size of the array. In such cases, the compiler automatically computes the size.

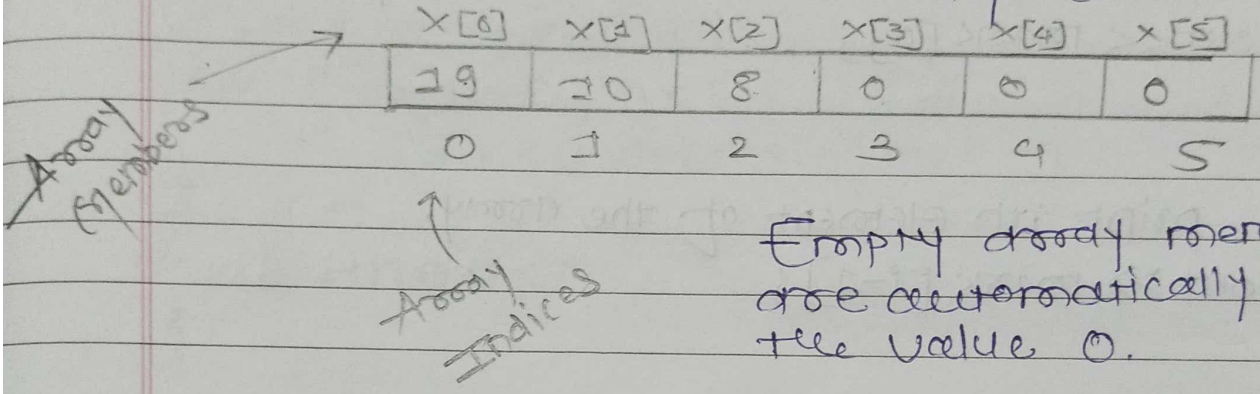
C++ Array With Empty Members

In C++, if an array has a size (n), we can store upto n number of elements in the array. However, what will happed if we store less than (n), number of element.

```
// store only 3 element in the array
int x[6] = {19, 10, 8};
```

Here, the array x has a size of 6. However, we have initilized it will only 3 elements.

In such cases, the compiler assigns random value to the remaining places. Oftentimes, this random value is simply 0.



Empty array members are automatically assigned the value 0.

How to Insert & print array Elements.

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

1. Change the fourth element to 9.

```
mark[3] = 9;
```

2. take the input from the user stored the value third position.

```
Cin>>marks[2];
```

Saathi

Date __/__/____

3. Take the input from the user insert at ith position.

```
Cin >> marc[i-1]
```

4. print first element of the array

```
Cout<< marc[0];
```

5. print ith element of the array.

```
Cout >> marc[i-1];
```

Quick Quiz:

Display the array element.

Example 2: Take Input from User & store them in an array.

```
#include<iostream>
using namespace std;

int main(){
    int numbers[5];

    Cout<<"Enter 5 numbers: "<<endl;
```

Date __/__/__

Saathi

```
// store input from user to array
for (int i = 0; i < 5; ++i) {
        cin >> numbers[i];
}
   cout << " The numbers are : ";

// print array elements
for (int n = 0; n < 5; ++n) {
        cout << numbers[n] << " ";
}
   return 0;
}
```