Content 12

Arrays and pointer arithmetics in C++.

#### What are Arrays in C++

* An array is a collection of items which are of the similar type stored in contiguous memory locations.
* Sometimes, a simple variable is not enough to hold all the data.
* For example, let’s say we want to store the marks of 2500 students; initializing 2500 different variable for this task is not feasible.
* To solve this problem, we can define an array with size 2500 that can hold the marks of all students.
* For example **int marks[2500];**

**Code:**

//\*\*\*\*\*\*\*\*\*\*\* General array\*\*\*\*\*\*\*\*\*\*\*\*

#include <iostream>

using namespace std;

int main()

{

    int arr[] = {2, 4, 6, 8};

    cout << arr[0] << endl;

    cout << arr[1] << endl;

    arr[2] = 455; // I can also change the valuess here

    cout << arr[2] << endl;

    cout << arr[3] << endl;

    return 0;

}

**Output:**

2

4

455

8

**Code for printing the values of Arays by Loops:**

// displaying the alaments of ararys uaing the loops.

#include <iostream>

using namespace std;

int main()

{

    int arr[] = {2, 4, 6, 8};

    for (int i = 0; i < 4; i++)

    {

        cout << arr[i] << endl;

    }

    cout<<"\n\nDisplaying Values in Array using While Loop"<<endl;

    int i=0;

    while (i<4)

    {

        cout << arr[i] << endl;

        i++;

    }

    i=0;

    cout<<"\n\nDisplaying Values using Do-While Loops"<<endl;

    do

    {

       cout << arr[i] << endl;

        i++;

    } while (i<4);

    return 0;

}

**Output;**

2

4

6

8

Displaying Values in Array using While Loop

2

4

6

8

Displaying Values using Do-While Loops

2

4

6

8

#### Pointers and Arrays:

Storing the address of an array into pointer is different than storing the address of a variable into the pointer because the name of the array is an address of the first index of an array. So to use ampersand "&" with the array name for assigning the address to a pointer is wrong.

* &Marks --> Wrong
* Marks --> address of the first block

**Code:**

//Acessing values using the pointers.

#include <iostream>

using namespace std;

int main()

{

    int arr[] = {2, 4, 6, 8};

    int \*ptr = arr;

    cout << "The elenments 1 of arrays by pointer is: " << \*ptr << endl;

    cout << "The elenments 2 of arrays by pointer is: " << \*(ptr + 1) << endl;

    cout << "The elenments 3 of arrays by pointer is: " << \*(ptr + 2) << endl;

    cout << "The elenments 4 of arrays by pointer is: " << \*(ptr + 3) << endl;

    return 0;

}

**Output:**

The elenments 1 of arrays by pointer is: 2

The elenments 2 of arrays by pointer is: 4

The elenments 3 of arrays by pointer is: 6

The elenments 4 of arrays by pointer is: 8

**Code2:**

#include <iostream>

using namespace std;

int main()

{

    int arr[] = {2, 4, 6, 8};

    int \*p = arr;

    cout << \*(p++) << endl; //first 2 is printed them it goes for increment(means on 3 but not printed)

    cout << \*(++p) << endl; //it was first incremented by uper line and then again incremented here and gives the value.

    return 0;

}

**Output:**

2

6