COMPUTER SCIENCE

Example 2: Take Inputs from User and Store Them in an Array

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

int main() {
    int numbers[5];

    cout << "Enter 5 numbers: " << endl;

    //  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;
}
```

Output

```
Enter 5 numbers: 
11
12
13
```

```
14
15
The numbers are: 11  12  13  14  15
```

Once again, we have used a `for` loop to iterate from `i = 0` to `i = 4`. In each iteration, we took an input from the user and stored it in `numbers[i]`.

Then, we used another `for` loop to print all the array elements.

Example 3: Display Sum and Average of Array Elements Using for Loop

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

int main() {

    // initialize an array without specifying size
    double numbers[] = {7, 5, 6, 12, 35, 27};

    double sum = 0;
    double count = 0;
    double average;

    cout << "The numbers are: ";

    //  print array elements
    // use of range-based for loop
    for (const double &n : numbers) {
        cout << n << "  ";

        //  calculate the sum
        sum += n;

        // count the no. of array elements
        ++count;
    }

    // print the sum
    cout << "\nTheir Sum = " << sum << endl;

    // find the average
    average = sum / count;
    cout << "Their Average = " << average << endl;

    return 0;
}
```

Output

```
The numbers are: 7  5  6  12  35  27
Their Sum = 92
Their Average = 15.3333
```

In this program:

1. We have initialized a `double` array named `numbers` but without specifying its size. We also declared three double variables `sum`, `count`, and `average`.

   Here, `sum =0` and `count = 0`.
2. Then we used a range based `for` loop to print the array elements. In each iteration of the loop, we add the current array element to `sum`.
3. We also increase the value of `count` by `1` in each iteration, so that we can get the size of the array by the end of the for loop.
4. After printing all the elements, we print the sum and the average of all the numbers. The average of the numbers is given by `average = sum / count;`

Note: We used a ranged `for` loop instead of a normal `for` loop.

A normal `for` loop requires us to specify the number of iterations, which is given by the size of the array.

But a ranged `for` loop does not require such specifications.

C++ Array Out of Bounds

If we declare an array of size 10, then the array will contain elements from index 0 to 9.

However, if we try to access the element at index 10 or more than 10, it will result in Undefined Behaviour.