



**វិទ្យាស្ថានបច្ចេកវិទ្យាកម្ពុជា**  
Institute of Technology of Cambodia

**TP-08**  
**Working with Array**  
**in C++**

Academic Year: 2022 - 2023

## 1. C++ Arrays

**Array** is a variable that can store multiple values of the same type.

To declare an array, define the variable type, specify the name of the array followed by **square brackets** and specify the number of elements it should store:

Elements	2	7	4	8	1	6
Index	0	1	2	3	4	5

Syntax :     **dataType** arrayName[**arraySize**];

Example:

```
string cars[4];  
string cars[4] = {"Volvo", "BMW", "Ford", "Mazda"};  
string cars[] = {"Volvo", "BMW", "Ford"};  
int myNum[3] = {10, 20, 30};
```

## 2. Displaying Array Elements

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

Example:

```
#include <iostream>  
using namespace std;  
  
int main() {  
    int numbers[5] = {7, 5, 6, 12, 35};  
  
    cout << "The numbers are: ";  
  
    // Printing array elements  
    // using range based for loop  
    for (const int &n : numbers) {  
        cout << n << " ";  
    }  
  
    cout << "\nThe numbers are: ";  
  
    // Printing array elements  
    // using traditional for loop  
    for (int i = 0; i < 5; ++i) {  
        cout << numbers[i] << " ";  
    }  
  
    return 0;  
}
```

### Output

```
The numbers are: 7 5 6 12 35  
The numbers are: 7 5 6 12 35
```

### 3. Inputs from User and Store in Array

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]**.

Example:

```
#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
```

#### 4. Sum and Average of Array Elements

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]**.

Example:

```
#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
```

**Problem1:**

Write a program in C++ to ask a user for 10 numbers and store in an array. Display all numbers in array.

----

**Problem2:**

Write a program in C++ to store 7 numbers in an array called a1. Create a new array call a2. Make a2 has the data as same as a1 (copy a1 to a2). Display numbers in a2 from back to forth.

Reverse an Array in C++							
Original Array	15	8	27	78	10	5	30
Reverse Array	30	5	10	78	27	8	15

----

E.g:   for(k...)  
          a2[k] = a1[k]

Store 7 number in a1 (a1: 6 4 5 9 0 1 3)  
Create a new array a2 (same data as a1)  
(a2: 6 4 5 9 0 1 3)

=> Output:  
      Display a2 (3 1 0 9 5 4 6)

----

**Problem3:**

Write a program in C++ to store 5 numbers in an array (ask input from user). Find sum of all numbers in array and also average. Display sum and average on screen.

----

E.g:   Input: 3 7 -1 9 7

=> Output:  
      Sum=25, Average=5.0

----

**Problem4:**

Write a program in C++ program to store data of 5 students. Each student has name, score, email and phone number. The program ask user for info of these 5 students. Display all students' information on screen.

----

E.g:   Input: John   75       john@gamil.com   012433443

=> Output:

```
#Student1
John
75
john@gmailcom
012
#Student2
Johnny
85
johny@gmailcom
011
```

.....

----

**Problem5:**

Same as problem#4. Find average score. The program display info of all students who got scores more than average.

----

```
Input data
Find average score (5 students, sum/5)
Display students who score>= average
```

----