Arrays are defined as the collection of similar types of data items stored at contiguous memory locations. An array is the simplest data structure where each data element can be randomly accessed by using its index number. Following are the important terms to understand the concept of Array.

* Element − Each item stored in an array is called an element.
* Index − Each location of an element in an array has a numerical index, which is used to identify the element.

## Array Representation





As per the above illustration, the following are the important points to be considered.

* Index starts with 0.
* Array length is 10 which means it can store 10 elements.
* Each element can be accessed via its index. For example, we can fetch an element at index 6 as 9.

## Basic Operations

Following are the basic operations supported by an array.

* Traverse − print all the array elements one by one.
* Insertion − Adds an element at the given index.
* Deletion − Deletes an element at the given index.
* Search − Searches an element using the given index or by the value.
* Update − Updates an element at the given index.

## Traverse Operation

This operation is to traverse through the elements of an array.

### Example

The following program traverses and prints the elements of an array:

it produces the following result:

### Output

## Insertion Operation

Insert operation is to insert one or more data elements into an array. Based on the requirement, a new element can be added at the beginning, end, or any given index of an array.

### Example

### Output

## Deletion Operation

Deletion refers to removing an existing element from the array and re-organizing all elements of an array.

### Example

### Output

## Search Operation

You can perform a search for an array element based on its value or its index.

### Example

### Output

## Update Operation

Update operation refers to updating an existing element from the array at a given index.

### Example

### Output

## The complexity of Array operations

Time and space complexity of various array operations are described in the following table.

### Time Complexity

| **Algorithm** | **Average Case** | **Worst Case** |
| --- | --- | --- |
| Access | O(1) | O(1) |
| Search | O(n) | O(n) |
| Insertion | O(n) | O(n) |
| Deletion | O(n) | O(n) |

### Space Complexity

In array, space complexity for the worst case is **O(n)**.

# 2D Array

2D array can be defined as an array of arrays. The 2D array is organized as matrices which can be represented as the collection of rows and columns. It produces a data structure that looks like the following.

The first element of the first row is represented by a[0][0] where the number shown in the first index is the number of that row while the number shown in the second index is the number of the column.