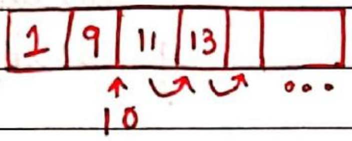
Insertion

An element can be inserted in an array at a specified position.

In order for this operation to be successful, the array should have enough capacity

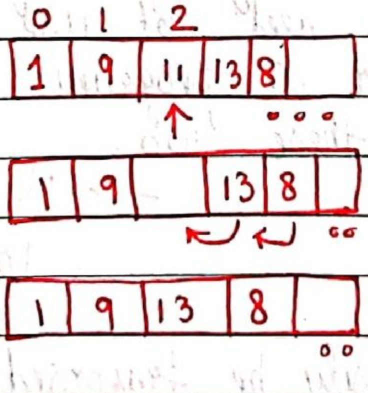


=> Elements need to be shifted to maintain relative order.

When no position is specified its best to insert the element at the end.

Deletion

An element at specified position can be deleted creating a void which needs to be fixed by shifting all the elements to the left as follows:



Delete 11 at ind 2

Shift the elements

Deletion done!

We can also bring the last element of the array to fill the void if the relative ordering is not important

Operations on an Array

Following operations are supported by an array.

Traversal
Insertion
Deletion
Search

⇒ There can be many other operations one can perform on arrays as well!
eg: Sorting asc., Sorting desc.

Traversal

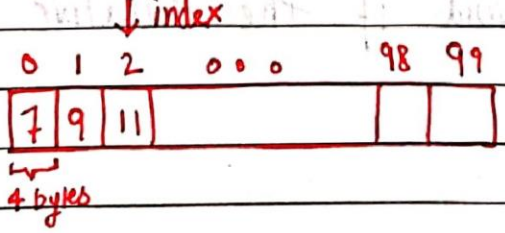
Visiting every element of an array once → Traversal

Why traversal? → For use cases like:
→ Storing all elements → using scanf
→ Printing all elements → using printf

An important note about arrays

If we create an array of length 100 using a[100] in C language, we need not use all the elements. It is possible for a program to use just 60 elements out of these 100.
↳ But we cannot go beyond 100 elements.

An array can easily be traversed using a for loop in C language



Searching

Searching can be done by traversing the array until the element to be searched is found

0	1	2	3	
7	9	11	12	...

→ Search

⇓

For sorted array time taken to search is much less than unsorted array!!

Sorting

Sorting means arranging an array in order (asc or desc)

We will see various sorting techniques later in the course.

12	7	18	1	8

unsorted array

⟹

1	7	8	12	18

sorted array