ACTIVITY-4

Abimanyu M EBEON0323767152

BATCH NO:2022-9615

SEARCHING ALGORITHMS

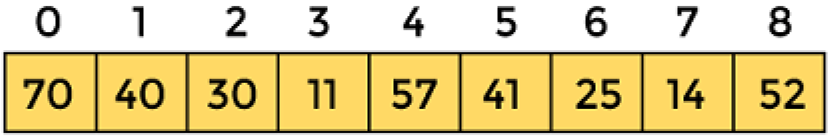
1, Linear Search Algorithm

Linear search is the simplest method for searching.
In Linear search technique of searching; the element is searched sequentially in the list.
This method can be performed on a sorted or an unsorted list (usually arrays).
In case of a sorted list searching starts from 0th element and continues until the element is found from the list or the element whose value is greater than (assuming the list is sorted in ascending order), the value being searched is reached.

Now, let's see the working of the linear search Algorithm.

To understand the working of linear search algorithm, let's take an unsorted array. It will be easy to understand the working of linear search with an example.

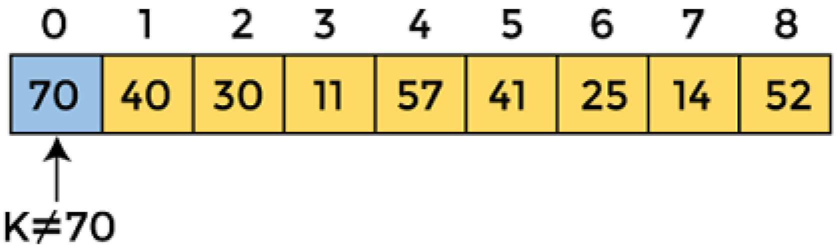
Let the elements of array are -



0	1	2	3	4	5	6	7	8
70	40	30	11	57	41	25	14	52

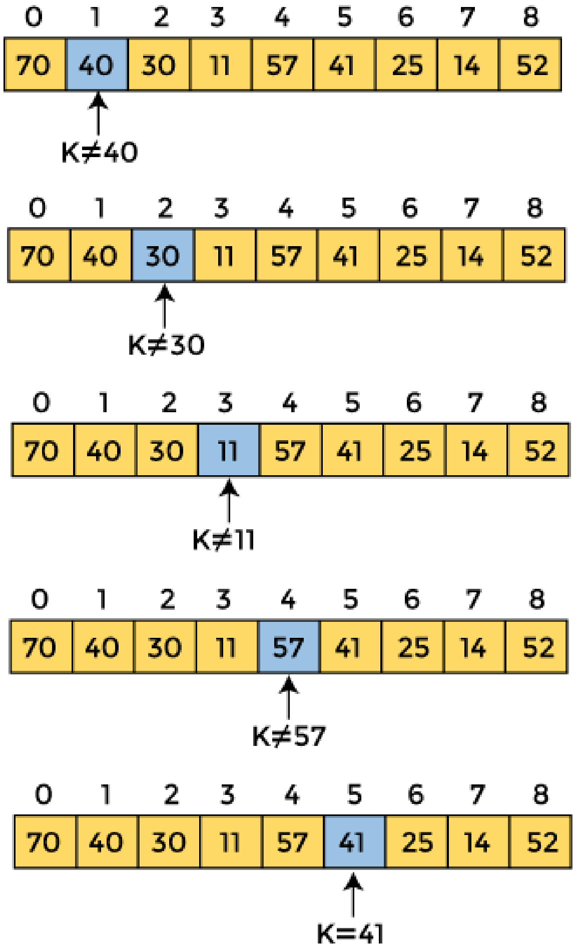
Let the element to be searched is **K = 41**

Now, start from the first element and compare **K** with each element of the array.



0	1	2	3	4	5	6	7	8
70	40	30	11	57	41	25	14	52

The value of **K,** i.e., **41,** is not matched with the first element of the array. So, move to the next element. And follow the same process until the respective element is found.



Now, the element to be searched is found. So algorithm will return the index of the element matched

2,Binary Search Algorithm

Binary search is **an efficient algorithm for finding an item from a sorted list of items**. It works by repeatedly dividing in half the portion of the list that could contain the item, until you've narrowed down the possible locations to just one.

Now, let's see the working of the Binary Search Algorithm.

To understand the working of the Binary search algorithm, let's take a sorted array. It will be easy to understand the working of Binary search with an example.

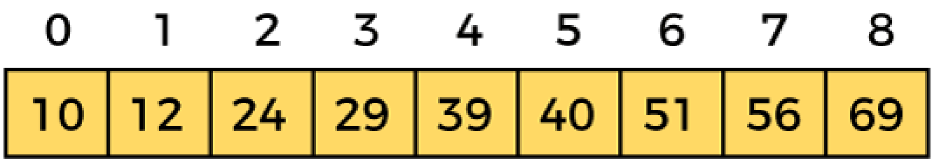
There are two methods to implement the binary search algorithm -

- Iterative method

- Recursive method

The recursive method of binary search follows the divide and conquer approach.

Let the elements of array are -



0	1	2	3	4	5	6	7	8
10	12	24	29	39	40	51	56	69

Let the element to search is, **K = 56**

We have to use the below formula to calculate the **mid** of the array -

1. mid = (beg + end)/2

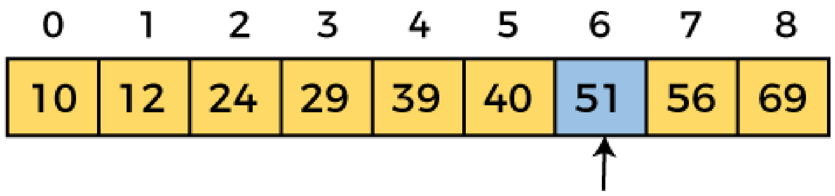
So, in the given array -

beg = 0

end = 8

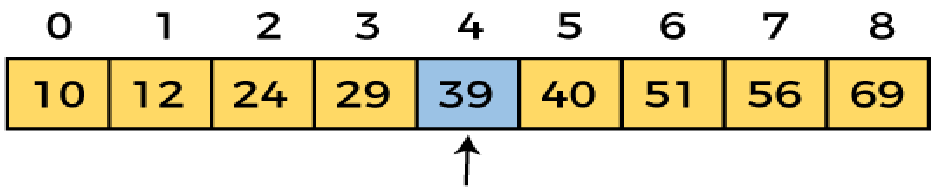
mid = (0 + 8)/2 = 4. So, 4 is the mid of th array.

e



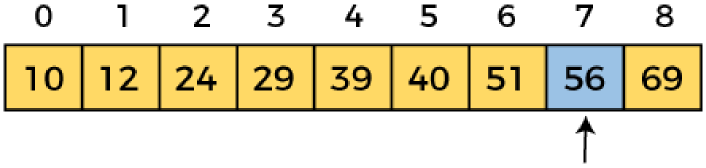
0	1	2	3	4	5	6	7	8
10	12	24	29	39	40	51	56	69

A[mid] = 51
A[mid] < K (or, 51 < 56)
So, beg = mid + 1 = 7, end = 8
Now, mid =(beg + end)/2 = 15/2 = 7



0	1	2	3	4	5	6	7	8
10	12	24	29	39	40	51	56	69

A[mid] = 39
A[mid] < K (or,39 < 56)
So, beg = mid + 1 = 5, end = 8
Now, mid =(beg + end)/2 = 13/2 = 6



A[mid] = 56
A[mid] = K (or, 56 = 56)
So, location = mid
Element found at 7th location of the array

Now, the element to search is found. So algorithm will return the index of the element matched.