**BINARY SEARCH ALGORITHM**

* It is a method to search whether any element is present in a sorted array, if it is present need to find in which index.
* It is a search method where we search a target element by taking the middle element and searching the target element in the left or right of the middle element.
* Here the time complexity is O (log N).

Consider a sorted array in ascending order,

the middle element is given by: mid = (first + last)/ 2

Consider a target element which is less than middle element

So need to check in the left side of the array and vice versa. If target element is equal to middle element, then the search come to end then and there.

In general,

To check the target element if the array is ascending:

1. If target element < middle element

first = remain same

last = mid – 1

1. If target element > middle element

first = mid + 1

last = remain same

To check the target element if the array is descending:

1. If target element < middle element

first = mid + 1

last = remain same

1. If target element > middle element

First = remain same

last = mid -1

**Time Complexity**

|  |
| --- |
|  |

0 N

1st comparison

|  |  |
| --- | --- |
|  |  |

0 N/2 = N/21 N

2nd comparison

|  |  |
| --- | --- |
|  |  |

N/4 = N/22

3rd comparison

|  |  |
| --- | --- |
|  |  |

N/8 = N/23

Kth comparison

|  |  |
| --- | --- |
|  |  |

N/2K

1= N/2K

2k = N

Log 2k = log N

k log 2 = log N

k = log N/log 2

k = log2 N

So the worst case scenario gives a time complexity of O (log N)

**To check whether the array is sorted descending or ascending**

Take first and last element of the array;

If first > last == descending order

If last > first == ascending order