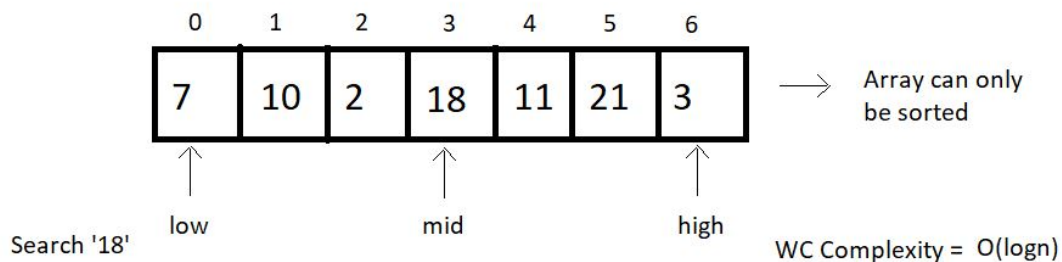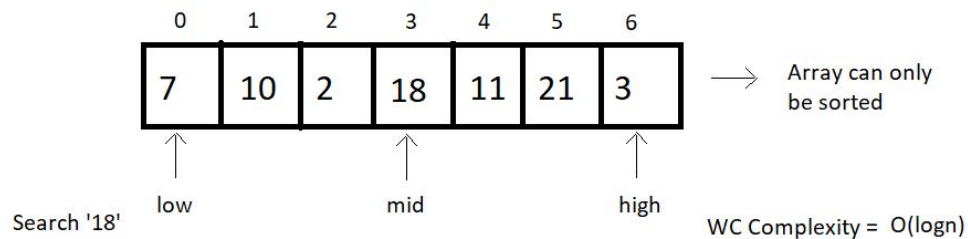# Binary Search Algorithm

This search method searches for an element by breaking the search space into half each time it finds the wrong element. This method is limited to a sorted array. The search continues towards either side of the mid, based on whether the element to be searched is lesser or greater than the mid element of the current search space.

0	1	2	3	4	5	6
7	10	2	18	11	21	3

→ Array can only be sorted

↑ low   ↑ mid   ↑ high

Search '18'

WC Complexity = O(logn)

0	1	2	3	4	5	6
7	10	2	18	11	21	3

→ Array can only be sorted

↑ low   ↑ mid   ↑ high

Search '18'

WC Complexity = O(logn)

```
int binarySearch(int arr[], int size, int element){
    int low, mid, high;
    low = 0;
    high = size-1;
    // Keep searching until low <= high
    while(low<=high){
        mid = (low + high)/2;
        if(arr[mid] == element){
```

```
            return mid;
        }
        if(arr[mid]<element){
            low = mid+1;
        }
        else{
            high = mid -1;
        }
    }
    return -1;

}
```