```
const func = () => {
let arr = [1,2,3,4,5,5,6,7];
let target = 6;
let low = 0;
let high = arr.length -1;
let mid;

 while(low<=high){
      mid = Math.floor(low+(high-low)/2);

      if(arr[mid] == target)
       {
          return mid;
       }

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

given a sorted array which migh contain some repeas=ting elements, //target .
   find out starting and ending index of the target.

```
let arr = [1,4,5,6,7,7,7,8,9,9];
let target = 7
//4,6

 let arr2 = [1,4,5,6,7,7,7,8,9,9];
let target2 = 3
//-1,-1

 let arr3 = [1,4,5,6,7,8,9,9];
let target3 = 7
//4,4

const func = (arr, target) => {
```

```
let n = arr.length;
let low = 0;
let high = n-1;
let mid;
let leftOcc = -1;
let rightOcc = -1;

// [1,4,5,6,7,7,7,7,7,7,8,9,9]
while(low<=high){
   mid = Math.floor(low+(high-low)/2);

   if(arr[mid] == target)
    {
       leftOcc = mid;
       high = mid-1;
    }

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

low = 0;
high = n-1;

while(low<=high){
   mid = Math.floor(low+(high-low)/2);

   if(arr[mid] == target)
    {
       rightOcc = mid;
       low = mid+1;
    }

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

```javascript
        }
    }

    console.log(leftOcc, rightOcc);
}

func(arr, target);
TC : O(logn)
SC: O(1)


 //Search in Rotated Sorted Array
 let arr1 = [8,9,1,2,3,4,5,7];//sorted array after rotation
 let target = 3
// There is an integer array nums sorted in ascending order (with distinct values).

const func =(arr, target) => {
    let n = arr.length;
    let low = 0;
    let high = n-1;
    let mid;
    while(low<=high){
        //[8,9,10,11,12,13,14,15,1,2,3,4,5,7] , target = 3
        mid = Math.floor(low+(high-low)/2);

        if(arr[mid] == target)
            return mid;

        else if(arr[low]<=arr[mid])
            {
                if(target < arr[mid] && target >= arr[low])
                {
                    high = mid -1;

                }
                else
                {
                    low = mid+1;
                }
            }

        else if(arr[mid]<=arr[high])
            {
                if(target > arr[mid] && target <= arr[high])
```

```
            {
                low = mid+1

            }
            else
            {
                high= mid-1;
            }
        }
    }

    return -1;//target is not present in array
}

console.log(func(arr1, target));
```