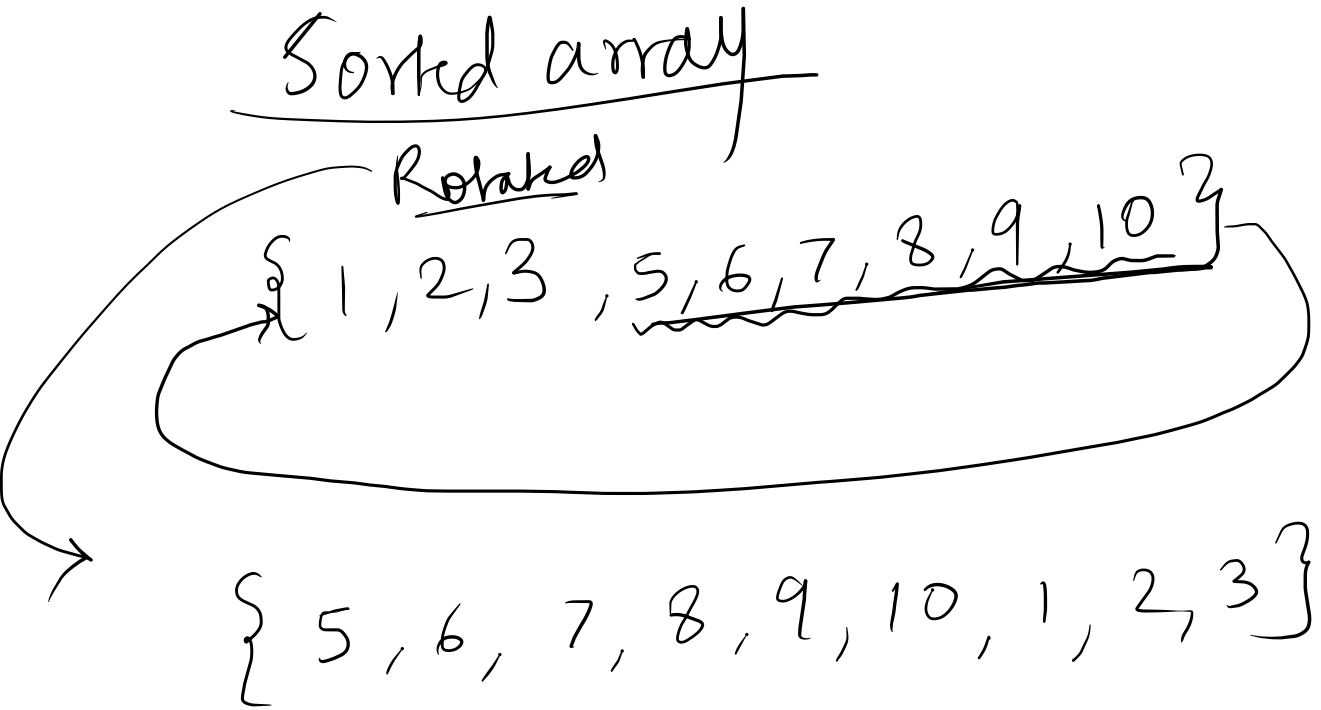
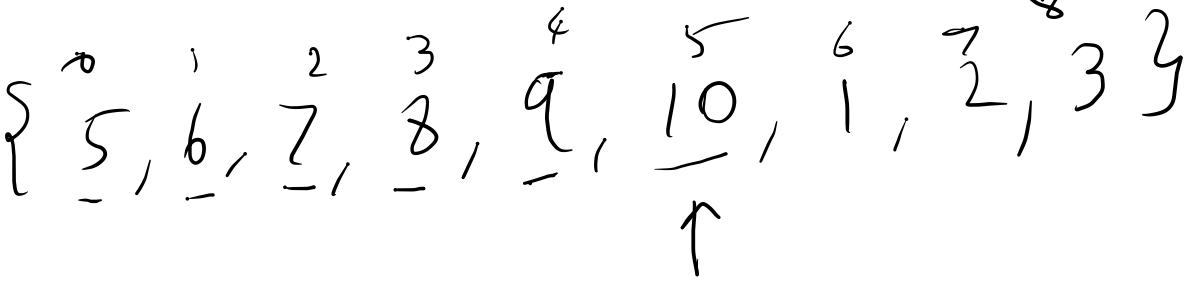
Divide & Conquer


Sorted array
Rotated
{1, 2, 3, 5, 6, 7, 8, 9, 10}
{5, 6, 7, 8, 9, 10, 1, 2, 3}

Rotated sorted array



Key = 10

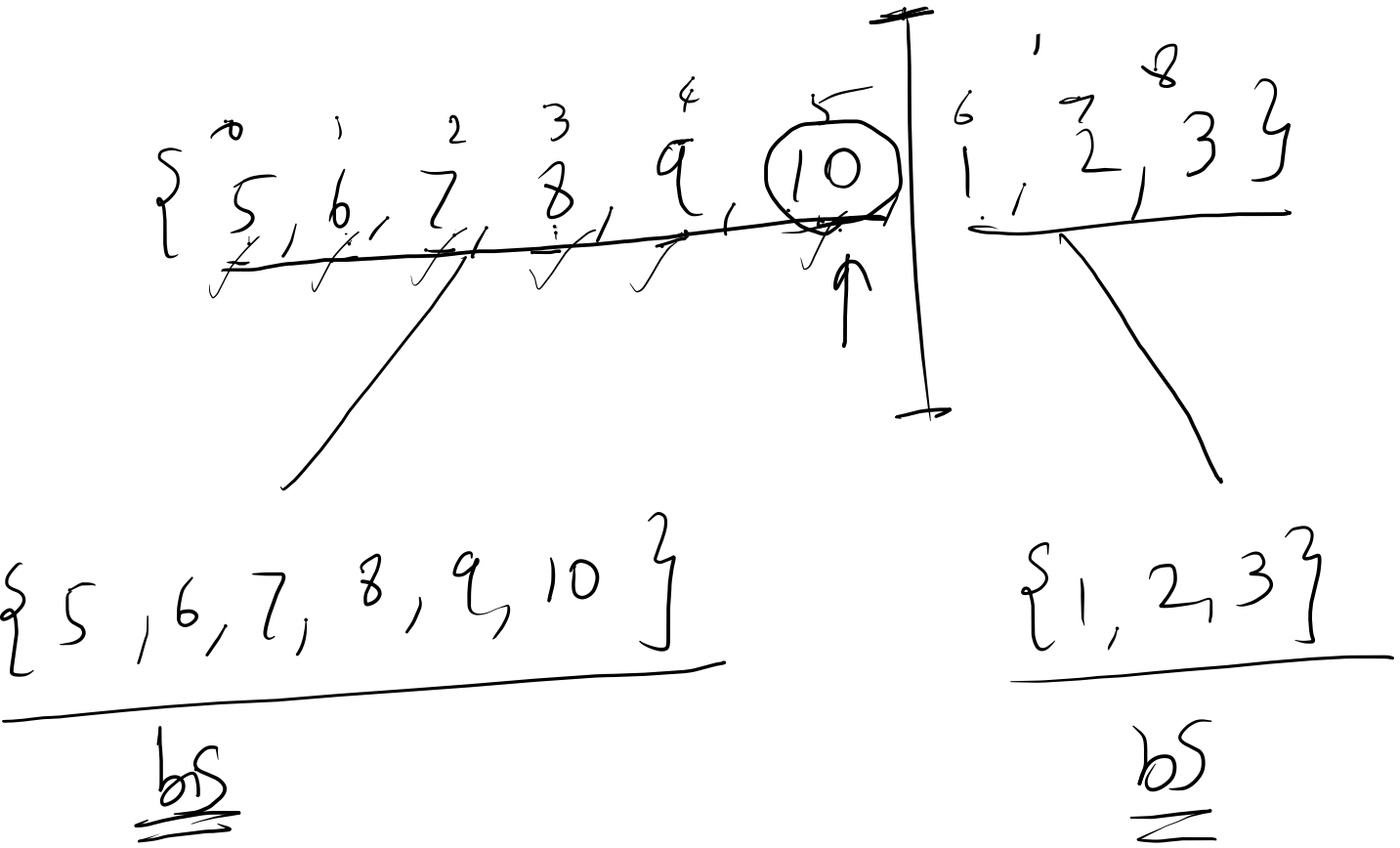
Linear search → O(N)

Rotated sorted array

0	1	2	3	4	5	6	7	8
5	6	7	8	9	10	1	2	3

Key = 10

binary search (0, n-1)

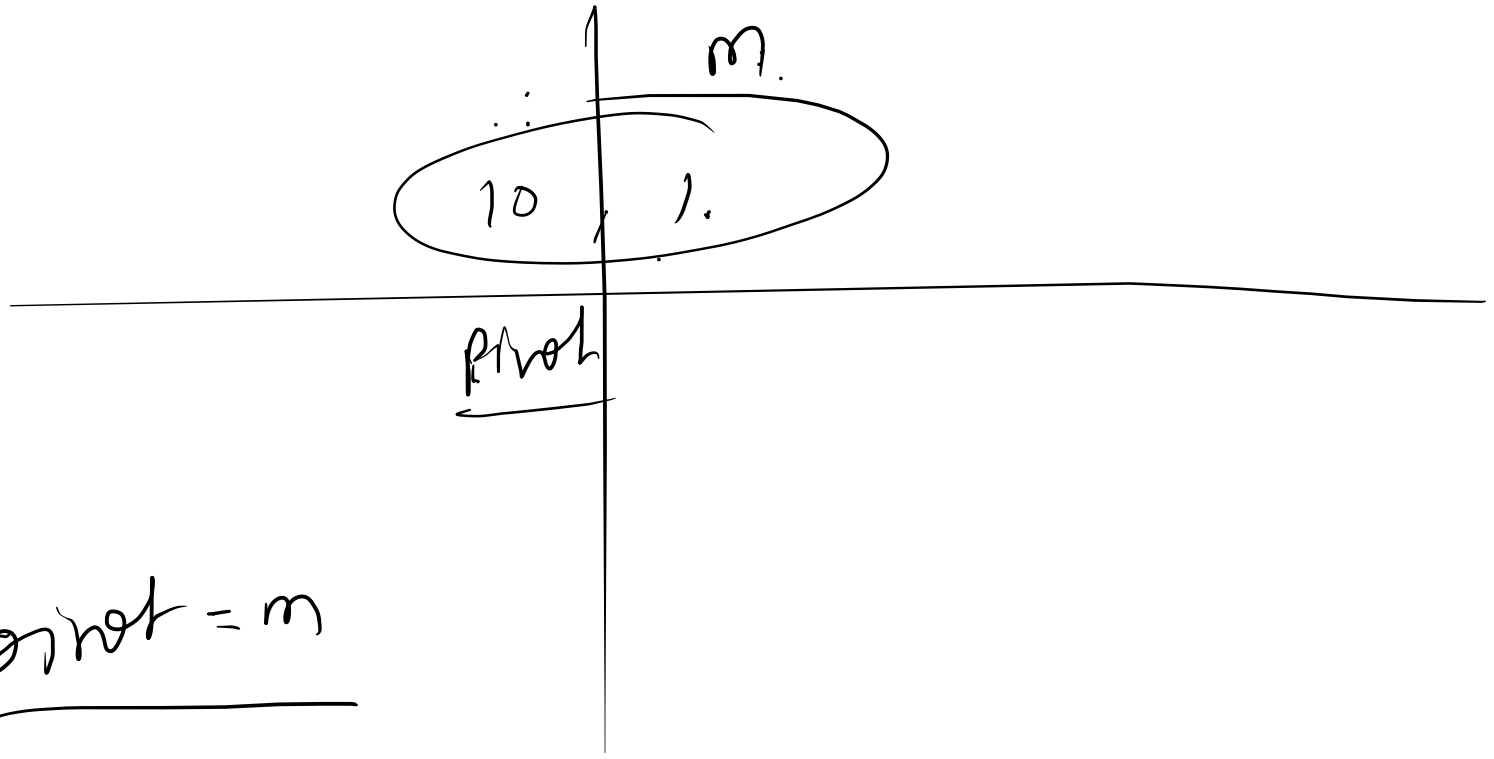

0 1 2 3 4 5 6 7 8
{5, 6, 7, 8, 9, 10}
{1, 2, 3}
{5, 6, 7, 8, 9, 10}
bs
{1, 2, 3}
bs

if (arr[m] > arr[m+1])

return m

if (arr[m-1] > arr[m])

return m-1



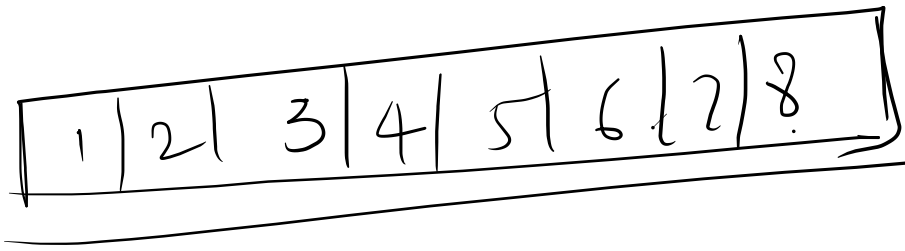
pivot = m

pivot = m - 1

$\{5, 6, 7, 8, 9, 10, 1, 2, 3\}$

$L = m + 1$

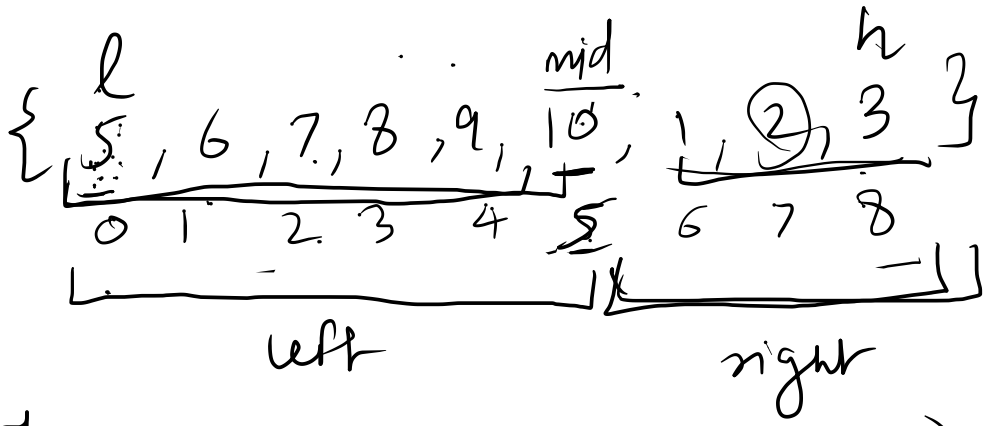
$R = m - 1$



return -1

Linear search → O(n)

✓ Binary search → O(log n)



key = 2

Key = 7

if (key >= arr [l])

l = mid

else

h = mid - 1

Key = 6

l m h

{ 5, 6, 7, 8, 9, 10, 1, 2, 3 }

0 1 2 3 4 5 6 7 8

m-1

arr[m] > arr[m+1]

1 > 2 ? X

(10, 1)

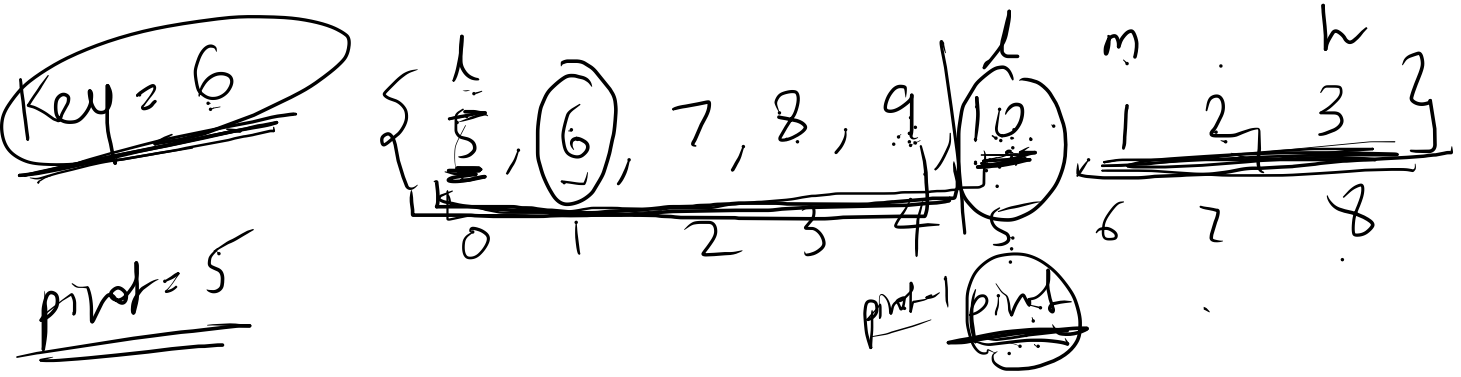
m-1 m

6

10 > 1 ?

arr[m-1] > arr[m]

return m-1



6 > 5

if (key >= arr[low]

h = prev-1

else

l = pivot+1

① find pivot

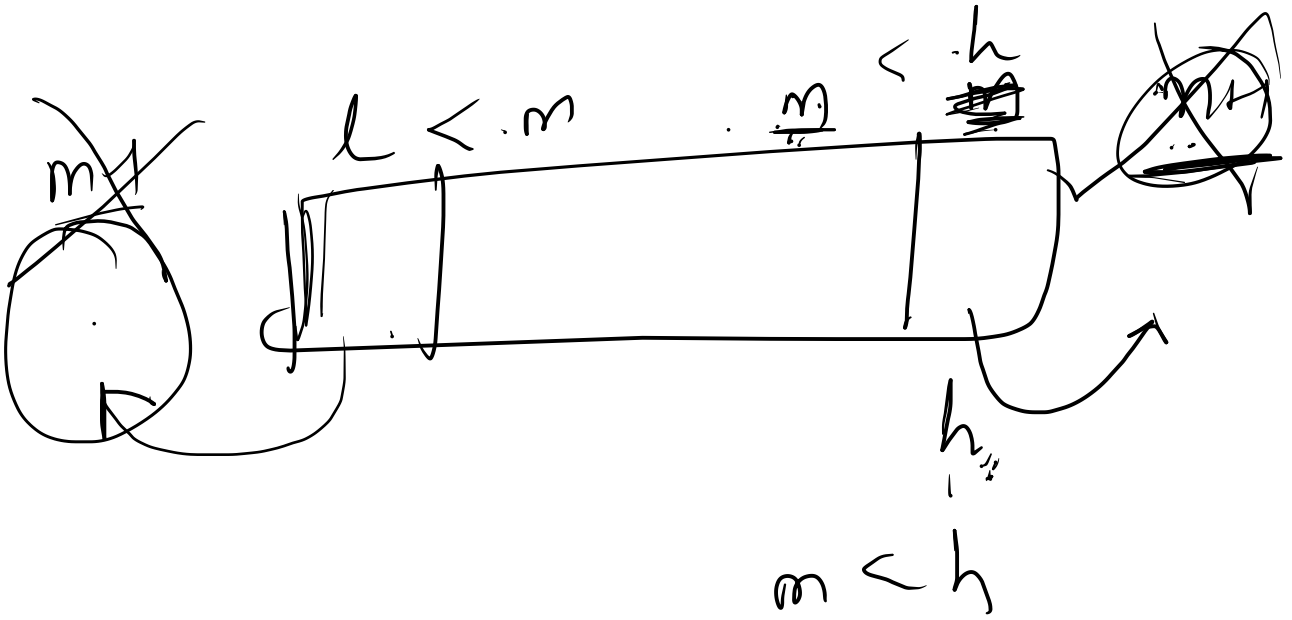
10 > 1

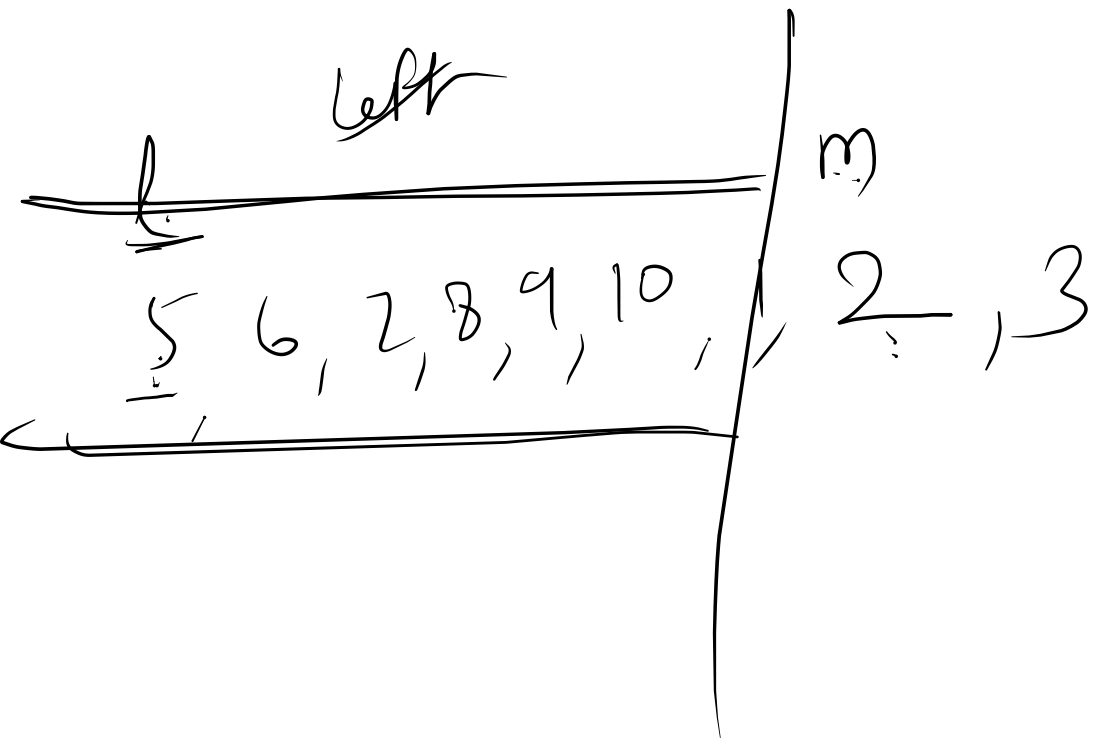
if (arr[m] > arr[m+1])

m-1 m

(10 , 1)

A[m-1] > A[m]


m
l < m
m
< h
h
m < h


left
m
5 6, 2, 8, 9, 10, 2, 3

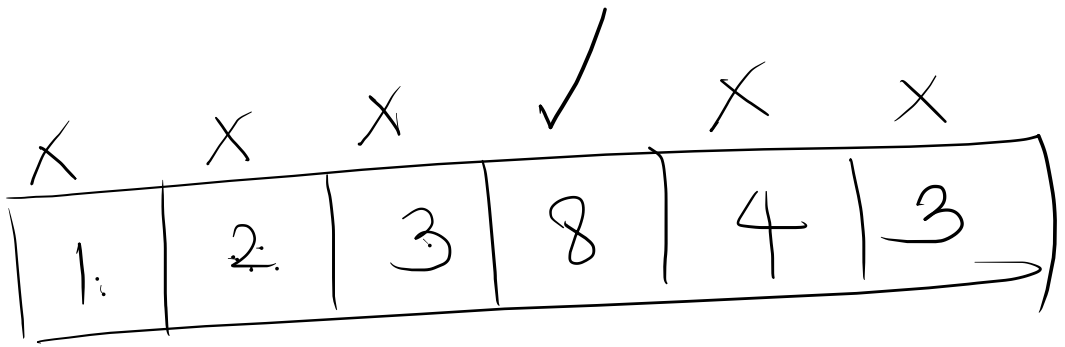
✓ Time complexity : $O(\log n)$

S.C = $O(1)$

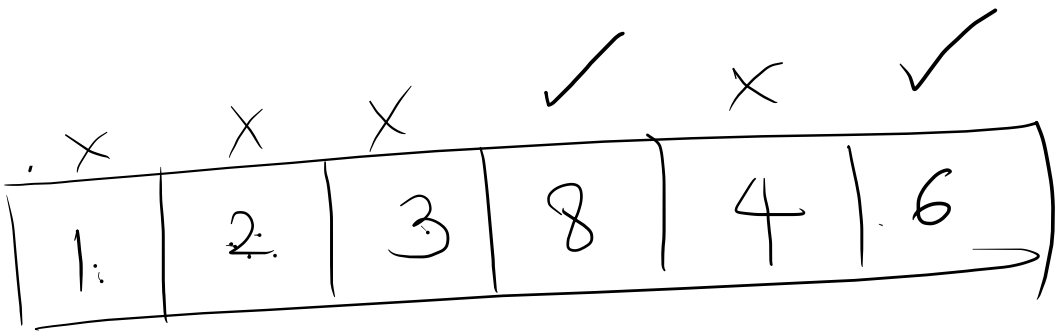
not smaller $\neq$ greater

not smaller = greater or equal

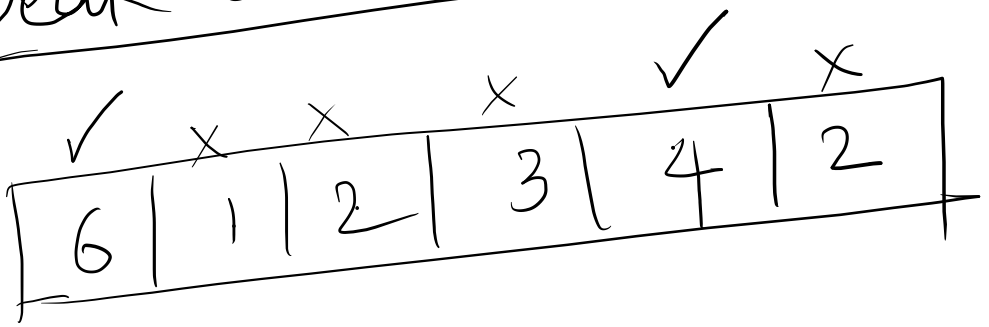
peak element



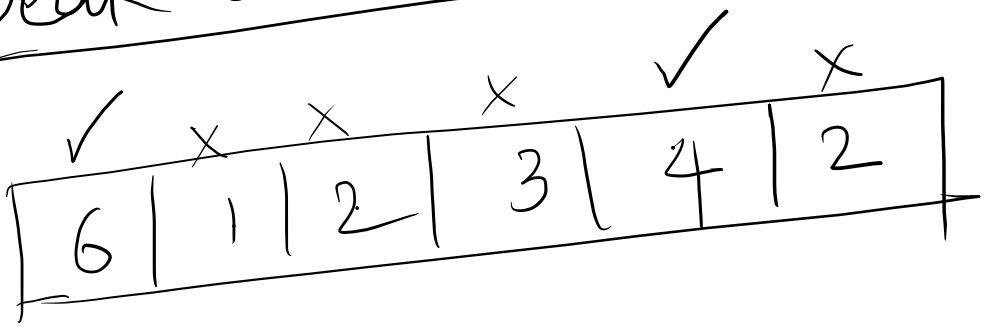
peak element



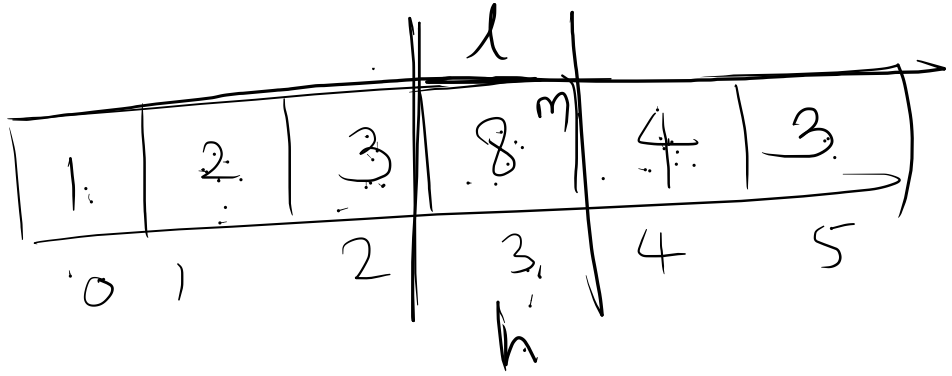
peak element



peak element



```
if (arr[m] >= arr[m-1] &&
    arr[m] >= arr[m+1])
    return m;
```



$m = \frac{l+h}{2} = \frac{3+5}{2} = 3$

if (arr[m] > left $8 > 3$ ✓

&& arr[m] > right) $8 > 4$ ✓

return m

① find pivot

if (arr [m-1] > arr[m])

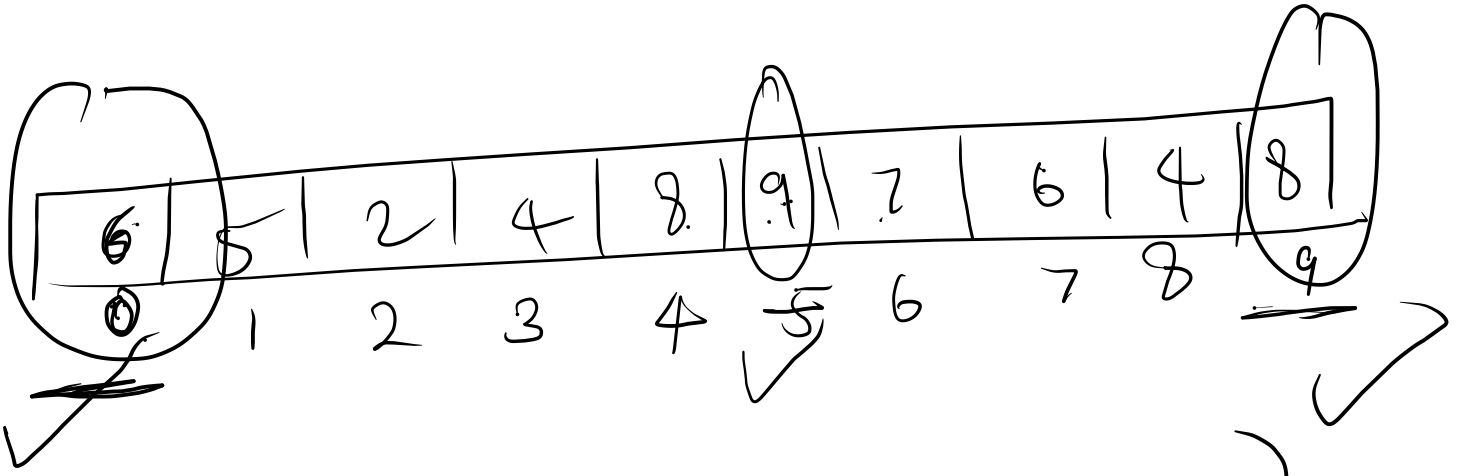
h = m - 1

if (arr [m+1] > arr [m])

L = m + 1

if (arr [m] > arr[m+1] && arr [m] > arr[m-1]

return m



① (arr[m] > left || m == 0)

&&

② (arr[m] > right || m == n-1)

Time complexity = $O(\log n)$