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

2 3 5 6 8 9 ~~10~~ → sorted array

10, 2, 3, 5, 6, 8, ~~9~~ → 1st rot.

9, 10, 2, 3, 5, 6, ~~8~~ → 2nd "

8, 9, 10, 2, 3, 5, 6 → 3rd rot.

Brute Force

Binary Search

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

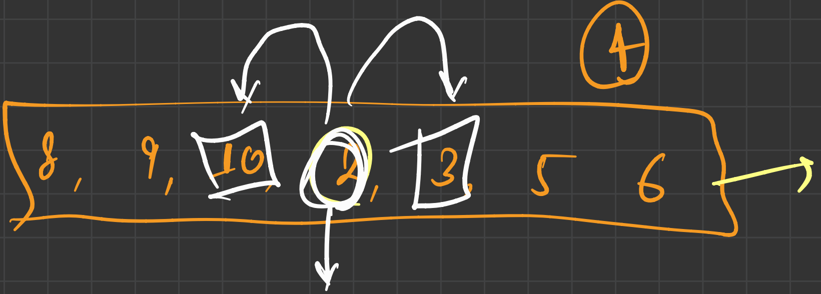
→ 3 Times

smallest element index

3

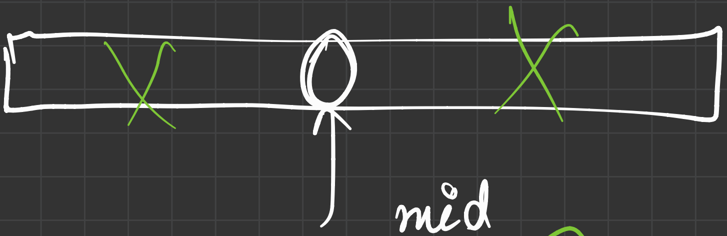
0	1	2	3	4	5	6	7
8	10	15	16	2	4	5	6

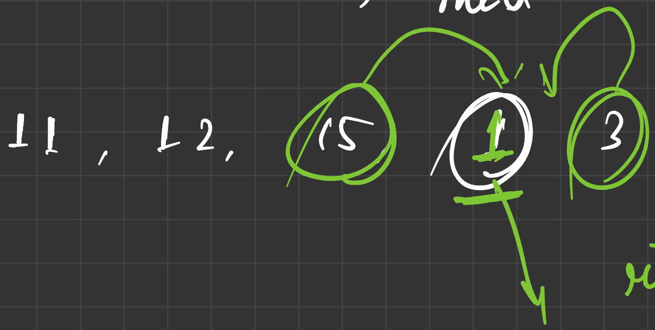
2, 4, 5, 6, 8, 10, 15, 16



B.S

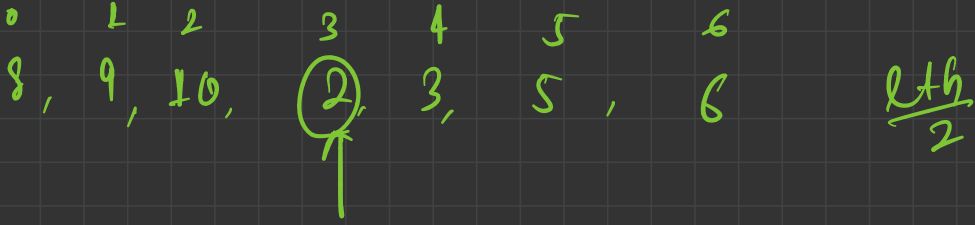
arr[mid] == target

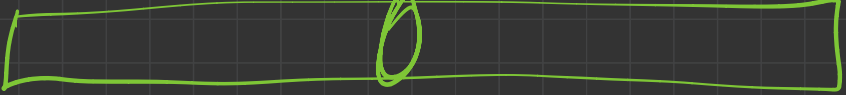




return mid;

```
if ( arr[mid] <= arr[prev] &&
         arr[mid] <= arr[next])
     return mid;
```





0	1	2	3	4	5	6	7
8,	9,	10,	12,	15,	2	3,	5

mid

$$\frac{7+0}{2} = \boxed{3}$$

mid

8, 9, 10, 12 ← 1st

12, 15, 2 3, 5 ← 2nd → unsorted array

0	1	2	3	4	5	6	7
6,	8,	10,	12,	15,	1,	2,	3

$$\frac{0+7}{2}$$

6, 8, 10, 12

12, 15, 1 2. 3

0	1	2	3	4	5	6	7	8
8	9,	10,	11,	12	2.	3,	5	6

$$m = \frac{0+8}{2}$$

if (arr[mid] <= arr[high])

```
        h= mid-1;
else if (arr [mid] >=  arr{ low ])
        l= mid + 1;
```