`nums[k-1]]` (**0-indexed**). For example, `[0,1,2,4,5,6,7]` might be rotated at pivot index `3` and become `[4,5,6,7,0,1,2]`.

Given the array `nums` **after** the rotation and an integer `target`, return *the index of* `target` *if it is in* `nums`, *or* `-1` *if it is not in* `nums`.

You must write an algorithm with `O(log n)` runtime complexity.

Example 1:

```
Input: nums = [4,5,6,7,0,1,2], target = 0
Output: 4
```

Example 2:

```
Input: nums = [4,5,6,7,0,1,2], target = 3
Output: -1
```

Example 3:

```
Input: nums = [1], target = 0
Output: -1
```

Constraints:

- `1 <= nums.length <= 5000`
- `-10^4 <= nums[i] <= 10^4`
- All values of `nums` are **unique**.
- `nums` is guaranteed to be rotated at some pivot.
- `-10^4 <= target <= 10^4`

Accepted 1,108,234 Submissions 3,017,285

Seen this question in a real interview before? Yes No

Companies

Related Topics

Similar Questions

```
1  func search(n
   int) int {
2      return bi
   len(nums)-1,
3  }
4
5  func binSearc
   target int, n
6      // final
7      if bot -
8          if nu
9              r
10         }
11         if nu
12             r
13         }
14         retur
15     }
16
17     mid := (t
18
19     regCaseL
   nums[mid] &&
   && target <=
20     pivCaseL
   nums[mid] &&
   || target >=
21     regCaseR
   nums[top] &&
   && target >=
22     pivCaseR
   nums[top] &&
   || target <=
23
24     if regCas
25         // go
26         retur
   binSearchRota
   target, nums)
27     }
```

Testcase Run Code Result

Wrong Answer Run

Your input `[4,5,6,`
`0`

Output `-1`
`-1`

Expected `4`
`-1`

Use Example Testcases Run