35. Search Insert Position

Easy 6455 368 Add to List Share

Given a sorted array of distinct integers and a target value, return the index if the target is found. If not, return the index where it would be if it were inserted in order.

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

Example 1:

```
Input: nums = [1,3,5,6], target = 5
Output: 2
```

Example 2:

```
Input: nums = [1,3,5,6], target = 2
Output: 1
```

Example 3:

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

Constraints:

- `1 <= nums.length <= 10^4`
- `-10^4 <= nums[i] <= 10^4`
- `nums` contains **distinct** values sorted in **ascending** order.
- `-10^4 <= target <= 10^4`

Accepted 1,208,852 Submissions 2,839,709

Seen this question in a real interview before? Yes No

Companies

Java Autocomplete

```java
class Solution {
    public int searchInsert(int[] nums, int target) {

        int length = nums.length;
        int start = 0;
        int end = nums.length - 1;

        while(start < length && nums[start] <= target){

            int middle = start + (end - start) / 2;

            if(nums[middle] == target)
                return middle;
            else if(nums[middle] < target)
                start = middle + 1;
            else{
                end = middle - 1;
            }
        }

        return start;
    }
}
```

Your previous code was restored from your local storage. Reset to default