

Problem 35: Search Insert Position

Problem Information

Difficulty: Easy

Acceptance Rate: 50.07%

Paid Only: No

Tags: Array, Binary Search

Problem Description

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:

$1 \leq \text{nums.length} \leq 10^4$ $-10^4 \leq \text{nums}[i] \leq 10^4$ nums contains **distinct** values sorted in **ascending** order. $-10^4 \leq \text{target} \leq 10^4$

Code Snippets

C++:

```
class Solution {  
public:  
    int searchInsert(vector<int>& nums, int target) {  
  
    }  
};
```

Java:

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

Python3:

```
class Solution:  
    def searchInsert(self, nums: List[int], target: int) -> int:
```