Binary Search

LeetCode 704 - Binary Search *[easy]*

Given a sorted (in ascending order) integer array `nums` of **n** elements and a `target` value, write a function to search `target` in `nums`. If `target` exists, then return its index, otherwise return **-1**.

Example 1:

```
Input: nums = [-1, 0, 3, 5, 9, 12], target = 9
Output: 4
Explanation: 9 exists in nums and its index is 4
```

Example 2:

```
Input: nums = [-1, 0, 3, 5, 9, 12], target = 2
Output: -1
Explanation: 2 does not exist in nums so return -1
```

Note:

- You may assume that all elements in `nums` are unique.
- **n** will be in the `range [1, 10000]`.
- The value of each element in `nums` will be in the `range [-9999, 9999]`.

```java
public int binarySearch(int[] nums, int key){
        int low = 0;
        int high = nums.length - 1;
        int mid = (low + high) / 2;
        while(low <= high ){
            if (nums[mid] == key){
                return nums[mid];
            }else if (nums[mid] < key){
                low = mid + 1;
            }else{
                high = mid - 1;
            }
        }
        return -1;
    }
```

LeetCode 744 - Find Smallest Letter Greater Than Target [*easy*]