

Find First and Last Position of Element in Sorted Array

Given an array of integers `nums` sorted in ascending order, find the starting and ending position of a given target value. Your algorithm's runtime complexity must be in the order of $O(\log n)$.

If the target is not found in the array, return `[-1, -1]`.

Example 1:

```
Input: nums = [5,7,7,8,8,10], target = 8
Output: [3,4]
```

Example 2:

```
Input: nums = [5,7,7,8,8,10], target = 6
Output: [-1, -1]
```

Solution in C++

```
class Solution {
public:
    vector<int> searchRange(vector<int>& nums, int target) {
        vector<int> r = {-1, -1};

        int n = nums.size();
        for (int i = 0; i < n; i++) {
            if (r[0] < 0) {
                if (nums[i] == target) r[0] = r[1] = i;
            }
            else if (nums[i] != target) break;
            else r[1] = i;
        }

        return r;
    }
};
```