The "Two-Pass" Approach

Wednesday, December 31, 2025 2:23 AM

1. The core conflict

A standard Binary Search has one goal: find ANY match.

- When it sees nums[mid]==target, it returns mid immediately.
- The Failure: If the array is [8,8,8], standard BS returns index 1. It doesn't know if indices 0 (or) 2 are also 8. We can't scan linearly (while...) because that degrades to O(n) in the worst case (e.g., all elements are identical).

2. The Solution

To crack this, we modify the behavior only when a match is found. Instead of stopping, we force the search to continue in a specific condition.

We run Binary Search twice:

1. Search A (Left Bias) : Finds the start index.
2. Search B (Right Bias): Finds the End Index.

3. Logic Breakdown

Here is exactly how the logic changes for the two searches.

Search A : Finding the first Position (Start)

- Goal: Find the target furthest to the left.
- Logic: when nums[mid] == target:
  - "I found a target at mid. This might be the first one, (or) maybe there's another one before it."
  - Action:
    1. Save mid as a potential answer.
    2. Reject the Right side. Move end=mid-1
    3. Continue searching the left half.

Search B : Finding the last Position (End)

- Goal: Find the target function to the right.
- Logic: when nums[mid] == target:
  - "I found a target at mid. This might be the last one, (or) maybe there's another one after it."
  - Action:
    1. Save mid as a potential answer.
    2. Reject the Left side. Move start=mid+1
    3. Continue searching the Right half.

4. Find Algorithm Steps

1. Check Edge Case: If the list is empty, return [-1,-1].
2. Run Left-Bias Search: Get start_index.
   - Optimization: If start_index is -1 (target not found), return [-1,-1] immediately. No need to run the second search.
3. Run Right-Bias Search : Get end_index
4. Return : [start_index, end_index]

5. Complexity Analysis

- Time: O(log n) + O(log n) ⇒ O(log n)
  We perform two independent binary searches.
- Space : O(1)
  We only store variables (start, end, mid, result).

SORTED ARRAYS GROUP DUPLICATES TOGETHER. IF YOU FIND ONE, THE OTHERS ARE IMMEDIATE NEIGHBORS.