Leetcode -167

167. Two Sum II - Input Array Is Sorted

Medium | Topics | Companies

Given a **1-indexed** array of integers `numbers` that is already ***sorted in non-decreasing order***, find two numbers such that they add up to a specific `target` number. Let these two numbers be `numbers[index₁]` and `numbers[index₂]` where `1 <= index₁ < index₂ <= numbers.length`.

Return *the indices of the two numbers,* `index₁` *and* `index₂`*,* ***added by one*** *as an integer array* `[index₁, index₂]` *of length 2.*

The tests are generated such that there is **exactly one solution**. You **may not** use the same element twice.

Your solution must use only constant extra space.

Example 1:

```
Input: numbers = [2,7,11,15], target = 9
Output: [1,2]
Explanation: The sum of 2 and 7 is 9. Therefore, index1 = 1, index2 = 2. We return [1, 2].
```

0	1	2	3	4	5	6
-4	1	2	7	10	11	14

target = 13

① Brute force: for each num look for the pair with the target sum.

-4 → 1, 2, 7, 10, 11, 14

1 → 2, 7, 10, 11, 14

i=2 2 → 7, 10, 11, 14 (j=5)

13

Time: $O(n^2)$

space: $O(1)$

② Better Approach – Binary Search → array sorted

0	1	2	3	4	5	6
-4	1	2	7	10	11	14

target = 13

13-(-4) = 17

13-2 = 11 → can be searched in logn time.

13-10=3 → search in left.

Search 17 in right (logn) time

Time: $O(n \log n)$

space: $O(1)$

③ More Better Approach

0	1	2	3	4	5	6
-4	1	2	7	10	11	14

target = 13

13-(-4)	13-1	13-2	13-7	13-10	13-11
17 ✗	12	11	6	3	2

[2, 5] → [3, 6]

Map

-4	0
1	1
2	2
7	3
10	4

Time: $O(n)$

Space: $O(n)$

③ Optimal Approach

target = 13

0	1	2	3	4	5	6
-4	1	2	7	10	11	14

l l l r r

arr[l] + arr[r] = -4 + 14 = 10 < 13 ⇒ l++

1 + 14 = 15 > 13 ⇒ r--

1 + 11 = 12 < 13 ⇒ l++

2 + 11 = $\boxed{13 = 13}$

Time: $O(n)$

Space: $O(1)$