Explore Problems(/problemset/all/) Mock(/interview/) Contest Discuss(/discuss/) Discuss(/discuss/) Discuss(/discuss/) Storedata=eyJ1cmwiOiAiaHR0cHM6Ly9sZWV0Y29kZS5jb20vam9icy8\_cmVmPX

# 5746. Minimum Distance to the Target Element

My Submissions (/contest/weekly-contest-239/problems/minimum-distance-to-the-target-element/submissions/)

Back to Contest (/contest/weekly-contest-239/)

Given an integer array nums (0-indexed) and two integers target and start, find an index i such that nums [i] == target and abs(i - start) is minimized. Note that abs(x) is the absolute value of x.

Return abs(i - start).

It is **guaranteed** that target exists in nums.

# User Accepted: 77 User Tried: 87 Total Accepted: 77 Total Submissions: 87 Difficulty: Easy

# Example 1:

```
Input: nums = [1,2,3,4,5], target = 5, start = 3
Output: 1
Explanation: nums[4] = 5 is the only value equal to target, so the answer is abs(4 - 3) = 1.
```

### Example 2:

```
Input: nums = [1], target = 1, start = 0
Output: 0
Explanation: nums[0] = 1 is the only value equal to target, so the answer is abs(0 - 0) = 1.
```

# Example 3:

```
Input: nums = [1,1,1,1,1,1,1,1,1,1], target = 1, start = 0
Output: 0
Explanation: Every value of nums is 1, but nums[0] minimizes abs(i - start), which is abs(0 - 0) = 0.
```

# Constraints:

- 1 <= nums.length <= 1000
- 1 <= nums[i] <=  $10^4$
- 0 <= start < nums.length
- target is in nums.

```
JavaScript
                                                                                                                              C
                                                                                                                         Ø
    const abs = Math.abs;
2
    const getMinDistance = (a, target, start) => {
3 ⋅
4
        let n = a.length;
 5
        let li, ri;
 6٠
        for (let i = start; i < n; i++) {
7 ▼
            if (a[i] == target) {
 8
                 ri = i;
9
                 break;
10
            }
11
12
        for (let i = start; \sim i; i--) {
13 ▼
             if (a[i] == target) {
14
                 li = i;
15
                 break;
16
            }
17
        // pr(ri, li);
18
        if (ri != undefined) {
19
20 🔻
            if (li != undefined) {
21
                 let ldiff = abs(li - start);
                 let rdiff = abs(ri - start);
22
                 // pr(ldiff, rdiff);
23
                 return ldiff <= rdiff ? ldiff : rdiff;</pre>
24
25 •
            } else {
26
                 return abs(ri - start);
```

```
Minimum Distance to the Target Element - LeetCode Contest
 27
                }
 28 ▼
           } else {
 29
                return abs(li - start);;
 30
31
      };
\ \square Custom Testcase
                        Use Example Testcases
                                                                                                                                     Run
                                                                                                                                                △ Submit
Submission Result: Accepted (/submissions/detail/487684967/) ?
                                                                                   More Details > (/submissions/detail/487684967/)
Share your acceptance!
Copyright © 2021 LeetCode
                          Help Center (/support) | Jobs (/jobs) | Bug Bounty (/bugbounty) | Students (/student) | Terms (/terms) | Privacy Policy (/privacy)
United States (/region)
```