# 1182. Shortest Distance to Target Color

My Submissions (/contest/biweekly-contest-8/problems/shortest-distance-to-target-color/submissions/)

Back to Contest (/contest/biweekly-contest-8/)

You are given an array colors, in which there are three colors: 1, 2 and 3.

You are also given some queries. Each query consists of two integers  $\mathtt{i}$  and  $\mathtt{c}$ , return the shortest distance between the given index  $\mathtt{i}$  and the target color  $\mathtt{c}$ . If there is no solution return  $\mathtt{-1}$ .

# User Accepted: 577 User Tried: 740 Total Accepted: 586 Total Submissions: 1362 Difficulty: Medium

### Example 1:

```
Input: colors = [1,1,2,1,3,2,2,3,3], queries = [[1,3],[2,2],[6
Output: [3,0,3]
Explanation:
The nearest 3 from index 1 is at index 4 (3 steps away).
The nearest 2 from index 2 is at index 2 itself (0 steps away)
The nearest 1 from index 6 is at index 3 (3 steps away).
```

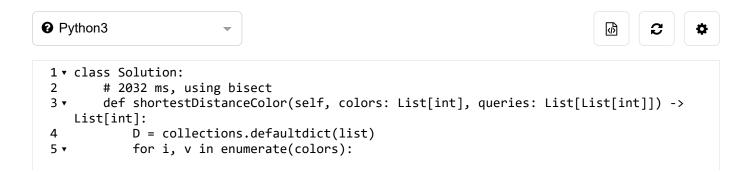
## Example 2:

```
Input: colors = [1,2], queries = [[0,3]]
Output: [-1]
Explanation: There is no 3 in the array.
```

### **Constraints:**

- 1 <= colors.length <= 5\*10^4
- 1 <= colors[i] <= 3
- 1 <= queries.length <= 5\*10^4
- queries[i].length == 2
- 0 <= queries[i][0] < colors.length
- 1 <= queries[i][1] <= 3

Discuss (https://leetcode.com/problems/shortest-distance-to-target-color/discuss)



1 of 2 9/8/2019, 2:18 AM

```
6
                 D[v].append(i)
 7
             print(D)
 8
             ans = []
 9 ▼
             for i, v in queries:
                 if v not in D:
10 ▼
                     ans.append(-1)
11
12
                     continue
13
                 index = bisect.bisect(D[v], i)
                 temp1 = D[v][index] if index < len(D[v]) else float("inf")</pre>
14
15
                 temp2 = D[v][index - 1] if index>0 else float("inf")
                 ans.append(min(abs(i - temp1), abs(i - temp2)))
16
17
             return ans
18
19
        # done in contest 2100 ms, search left and right, naive
20 ▼
        def shortestDistanceColor1(self, colors: List[int], queries: List[List[int]]) ->
    List[int]:
21
             def nearest_left(index, val):
22 ▼
23 ▼
                 for i in reversed(range(0, index +1)):
24 ▼
                     if colors[i] == val:
25
                         return index - i
26
                 return -1
27
28 ▼
             def nearest right(index, val):
29 ▼
                 for i in range(index, len(colors)):
30 ▼
                     if colors[i] == val:
                         return i - index
31
32
                 return -1
33
34
             ans = []
35
             D = \{\}
36 ▼
             for i, val in queries:
37 ▼
                 if (i, val) in D:
                     ans.append(D[(i,val)])
38
39
                     continue
                 left = nearest_left(i, val)
40
                 right = nearest_right(i, val)
41
42 ▼
                 if -1 in (left, right):
43
                     rv = max(left, right)
44 ▼
                 else:
45
                     rv = min(left, right)
46
                 ans.append(rv)
47
                 D[(i,val)] = rv
48
             return ans
```

☐ Custom Testcase

2 of 2 9/8/2019, 2:18 AM