C—(/) Explore(/explore/) Problems(/problemset/all/) Interview Contest Discuss(/discuss/) Discuss(/discuss/)

# 6284. Make Number of Distinct Characters Equal

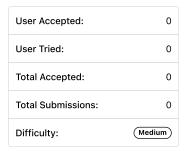
 $My\ Submissions\ (\ / contest/weekly-contest-327/problems/make-number-of-distinct-characters-equal/submissions/)$ 

Back to Contest (/contest/weekly-contest-327/)

You are given two **0-indexed** strings word1 and word2.

A **move** consists of choosing two indices i and j such that  $0 \le i \le word1$ .length and  $0 \le j \le word2$ .length and swapping word1[i] with word2[j].

Return true if it is possible to get the number of distinct characters in word1 and word2 to be equal with exactly one move. Return false otherwise.



#### Example 1:

```
Input: word1 = "ac", word2 = "b"
Output: false
Explanation: Any pair of swaps would yield two distinct characters in the first string, and one in the second string.
```

### Example 2:

```
Input: word1 = "abcc", word2 = "aab"
Output: true
Explanation: We swap index 2 of the first string with index 0 of the second string. The resulting strings are word1 = "abac" an
```

### Example 3:

```
Input: word1 = "abcde", word2 = "fghij"
Output: true
Explanation: Both resulting strings will have 5 distinct characters, regardless of which indices we swap.
```

# **Constraints:**

- 1 <= word1.length, word2.length <= 10<sup>5</sup>
- word1 and word2 consist of only lowercase English letters.

```
JavaScript
                                                                                                                          4
                                                                                                                                2
    const ord = (c) => c.charCodeAt();
1
    const isItPossible = (s, t) \Rightarrow \{
3 🔻
        let a = Array(26).fill(0), b = Array(26).fill(0);
 4
 5
        for (const c of s) a[ord(c) - 97]++;
        for (const c of t) b[ord(c) - 97]++;
 6
 7 ,
        for (let i = 0; i < 26; i++) {
 8 •
             if (a[i] > 0) {
9 ▼
                 for (let j = 0; j < 26; j++) {
10 ▼
                      if (b[j] > 0) {
                        a[i]--;
11
12
                        b[i]++;
13
                        b[j]--;
14
                        a[j]++;
15
                         if (equalDistinct(a, b)) return true;
                        a[i]++;
16
17
                        b[i]--;
18
                        b[j]++;
19
                        a[j]--;
20
21
                 }
22
             }
23
24
        return false;
25
    };
26
```

```
1/7/23, 10:13 PM
                                                               Make Number of Distinct Characters Equal - LeetCode Contest
   27 \bullet const equalDistinct = (a, b) => {
             let cntA = 0, cntB = 0;
for (let i = 0; i < 26; i++) {
   29 ▼
                  if (a[i] > 0) cntA++;
   30
                  if (b[i] > 0) cntB++;
   31
   32
   33
             return cntA == cntB;
   34
        };
  ☐ Custom Testcase
                          Use Example Testcases
                                                                                                                                       Run
                                                                                                                                                  Submission Result: Accepted (/submissions/detail/873752709/) ?
                                                                                     More Details > (/submissions/detail/873752709/)
  Share your acceptance!
   Copyright © 2023 LeetCode
  Help Center (/support) | Jobs (/jobs) | Bug Bounty (/bugbounty) | Online Interview (/interview/) | Students (/student) | Terms (/terms) | Privacy Policy (/privacy)
   United States (/region)
```