243. Shortest Word Distance

Easy 873 72 Add to List Share

Given an array of strings `wordsDict` and two different strings that already exist in the array `word1` and `word2`, return *the shortest distance between these two words in the list*.

Example 1:

```
Input: wordsDict = ["practice", "makes", "perfect", "coding", "makes"], word1 = "coding", word2 = "practice"
Output: 3
```

Example 2:

```
Input: wordsDict = ["practice", "makes", "perfect", "coding", "makes"], word1 = "makes", word2 = "coding"
Output: 1
```

Constraints:

- `1 <= wordsDict.length <= 3 * 10^4`
- `1 <= wordsDict[i].length <= 10`
- `wordsDict[i]` consists of lowercase English letters.
- `word1` and `word2` are in `wordsDict`.
- `word1 != word2`

Accepted 154,105 | Submissions 242,642

Seen this question in a real interview before? Yes No

Companies

0 ~ 6 months 6 months ~ 1 year 1 year ~ 2 years

LinkedIn | 15 Amazon | 5 Goldman Sachs | 3 Indeed | 2 Uber | 2 Salesforce | 2 DiDi | 2

Related Topics

Array String

Similar Questions

Shortest Word Distance II Medium

Shortest Word Distance III Medium

Java Autocomplete

```java
class Solution {
    public int shortestDistance(String[] words, String word1, String word2) {
        int i1 = -1, i2 = -1;
        int minDistance = words.length;
        for (int i = 0; i < words.length; i++) {
            if (words[i].equals(word1)) {
                i1 = i;
            } else if (words[i].equals(word2)) {
                i2 = i;
            }

            if (i1 != -1 && i2 != -1) {
                minDistance = Math.min(minDistance, Math.abs(i1 - i2));
            }
        }
        return minDistance;
    }
}
```