

[Explore](#)[Problems](#)[Interview](#)

New

[Contest](#)[Discuss](#)[Store](#)[Description](#)[Solution](#)[Discuss \(325\)](#)[Submissions](#)[Java](#)

244. Shortest Word Distance II

Medium

797

243

Add to List

Share

Design a data structure that will be initialized with a string array, and then it should answer queries of the shortest distance between two different strings from the array.

Implement the `WordDistance` class:

- `WordDistance(String[] wordsDict)` initializes the object with the strings array `wordsDict`.
- `int shortest(String word1, String word2)` returns the shortest distance between `word1` and `word2` in the array `wordsDict`.

Example 1:

Input

```
["WordDistance", "shortest", "shortest"]  
[[["practice", "makes", "perfect", "coding", "makes"]],  
 ["coding", "practice"], ["makes", "coding"]]
```

Output

```
[null, 3, 1]
```

Explanation

```
WordDistance wordDistance = new WordDistance(["practice",  
 "makes", "perfect", "coding", "makes"]);  
wordDistance.shortest("coding", "practice"); // return 3  
wordDistance.shortest("makes", "coding");    // return 1
```

Constraints:

- $1 \leq \text{wordsDict.length} \leq 3 \times 10^4$
- $1 \leq \text{wordsDict}[i].\text{length} \leq 10$
- `wordsDict[i]` consists of lowercase English letters.
- `word1` and `word2` are in `wordsDict`.
- `word1` \neq `word2`
- At most 5000 calls will be made to `shortest`.

Accepted 120,210

Submissions 203,781

[Problems](#)[Pick One](#)[Prev](#)

244/2199

[Next](#)[Run Code](#)

Your previous code

Console