Problem 243: Shortest Word Distance

Problem Information

Difficulty: Easy
Acceptance Rate: 66.12%
Paid Only: Yes
Tags: Array, String

Problem Description

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:

* `2 <= wordsDict.length <= 3 * 104` * `1 <= wordsDict[i].length <= 10` * `wordsDict[i]` consists of lowercase English letters. * `word1` and `word2` are in `wordsDict`. * `word1 != word2`

Code Snippets

C++:

```
class Solution {
public:
int shortestDistance(vector<string>& wordsDict, string word1, string word2) {
```

```
}
};
```

Java:

```
class Solution {
public int shortestDistance(String[] wordsDict, String word1, String word2) {

}
}
```

Python3:

```
class Solution:
def shortestDistance(self, wordsDict: List[str], word1: str, word2: str) ->
int:
```