Problem 737: Sentence Similarity II

Problem Information

Difficulty: Medium
Acceptance Rate: 50.94%
Paid Only: Yes
Tags: Array, Hash Table, String, Depth-First Search, Breadth-First Search, Union Find

Problem Description

We can represent a sentence as an array of words, for example, the sentence `"I am happy with leetcode"` can be represented as `arr = ["I","am",happy","with","leetcode"]`.

Given two sentences `sentence1` and `sentence2` each represented as a string array and given an array of string pairs `similarPairs` where `similarPairs[i] = [xi, yi]` indicates that the two words `xi` and `yi` are similar.

Return `true` _if`sentence1` and `sentence2` are similar, or _`false` _if they are not similar_.

Two sentences are similar if:

* They have **the same length** (i.e., the same number of words) * `sentence1[i]` and `sentence2[i]` are similar.

Notice that a word is always similar to itself, also notice that the similarity relation is transitive. For example, if the words `a` and `b` are similar, and the words `b` and `c` are similar, then `a` and `c` are **similar**.

Example 1:

Input: sentence1 = ["great","acting","skills"], sentence2 = ["fine","drama","talent"], similarPairs = [["great","good"],["fine","good"],["drama","acting"],["skills","talent"]] **Output:** true **Explanation:** The two sentences have the same length and each word i of sentence1 is also similar to the corresponding word in sentence2.

Example 2:

Input: sentence1 = ["I","love","leetcode"], sentence2 = ["I","love","onepiece"], similarPairs = [["manga","onepiece"],["platform","anime"],["leetcode","platform"],["anime","manga"]] **Output:** true **Explanation:** "leetcode" --> "platform" --> "anime" --> "manga" --> "onepiece". Since "leetcode is similar to "onepiece" and the first two words are the same, the two sentences are similar.

Example 3:

Input: sentence1 = ["I","love","leetcode"], sentence2 = ["I","love","onepiece"], similarPairs = [["manga","hunterXhunter"],["platform","anime"],["leetcode","platform"],["anime","manga"]] **Output:** false **Explanation:** "leetcode" is not similar to "onepiece".

Constraints:

* `1 <= sentence1.length, sentence2.length <= 1000` * `1 <= sentence1[i].length, sentence2[i].length <= 20` * `sentence1[i]` and `sentence2[i]` consist of lower-case and upper-case English letters. * `0 <= similarPairs.length <= 2000` * `similarPairs[i].length == 2` * `1 <= xi.length, yi.length <= 20` * `xi` and `yi` consist of English letters.

Code Snippets

C++:

```
class Solution {
public:
bool areSentencesSimilarTwo(vector<string>& sentence1, vector<string>&
sentence2, vector<vector<string>>& similarPairs) {

}
};
```

Java:

```
class Solution {
public boolean areSentencesSimilarTwo(String[] sentence1, String[] sentence2,
List<List<String>> similarPairs) {

}
}
```

Python3:

```
class Solution:
def areSentencesSimilarTwo(self, sentence1: List[str], sentence2: List[str],
similarPairs: List[List[str]]) -> bool:
```