734. Sentence Similarity

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] = [x i , y i ]` indicates that the two words `x i` and `y i` 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 not transitive. For example, if the words `a` and `b` are similar, and the words `b` and `c` are similar, `a` and `c` are **not necessarily similar**.

Example 1:

```
Input: sentence1 = ["great","acting","skills"], sentence2 = ["fine","drama","talent"], similarPairs = [["great","fine"],["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 = ["great"], sentence2 = ["great"], similarPairs = []
Output: true
Explanation: A word is similar to itself.
```

Example 3:

```
Input: sentence1 = ["great"], sentence2 = ["doubleplus","good"], similarPairs = [["great","doubleplus"]]
Output: false
Explanation: As they don't have the same length, we return false.
```

Constraints:

- `1 <= sentence1.length, sentence2.length <= 1000`
- `1 <= sentence1[i].length, sentence2[i].length <= 20`
- `sentence1[i]` and `sentence2[i]` consist of English letters.
- `0 <= similarPairs.length <= 1000`
- `similarPairs[i].length == 2`
- `1 <= x i .length, y i .length <= 20`
- `x i` and `y i` consist of lower-case and upper-case English letters.
- All the pairs `(x i , y i )` are **distinct**.