1657. Determine if Two Strings Are Close

Description

Two strings are considered **close** if you can attain one from the other using the following operations:

- Operation 1: Swap any two **existing** characters.
  - For example, `a b cd e -> a e cd b`
- Operation 2: Transform **every** occurrence of one **existing** character into another **existing** character, and do the same with the other character.
  - For example, `aa c abb -> bb c baa` (all `a`'s turn into `b`'s, and all `b`'s turn into `a`'s)

You can use the operations on either string as many times as necessary.

Given two strings, `word1` and `word2`, return `true` *if* `word1` *and* `word2` *are* ***close****, and* `false` *otherwise.*

Example 1:

```
Input: word1 = "abc", word2 = "bca"
Output: true
Explanation: You can attain word2 from word1 in 2 operations.
Apply Operation 1: "a bc" -> "a cb"
Apply Operation 1: "a c b" -> "b c a"
```

Example 2:

```
Input: word1 = "a", word2 = "aa"
Output: false
Explanation: It is impossible to attain word2 from word1, or vice versa, in any number of operations.
```

Example 3:

```
Input: word1 = "cabbba", word2 = "abbccc"
Output: true
Explanation: You can attain word2 from word1 in 3 operations.
Apply Operation 1: "ca b bb a" -> "ca a bb b"
Apply Operation 2: " c aa bbb" -> " b aa ccc"
Apply Operation 2: " baa ccc" -> " abb ccc"
```

Constraints:

- `1 <= word1.length, word2.length <= 10^5`
- `word1` and `word2` contain only lowercase English letters.