Problem 2840: Check if Strings Can be Made Equal With Operations II

Problem Information

Difficulty: Medium
Acceptance Rate: 55.87%
Paid Only: No
Tags: Hash Table, String, Sorting

Problem Description

You are given two strings `s1` and `s2`, both of length `n`, consisting of **lowercase** English letters.

You can apply the following operation on **any** of the two strings **any** number of times:

* Choose any two indices `i` and `j` such that `i < j` and the difference `j - i` is **even** , then **swap** the two characters at those indices in the string.

Return `true` _if you can make the strings_`s1` _and_`s2` _equal, and _`false` _otherwise_.

Example 1:

Input: s1 = "abcdba", s2 = "cabdab" **Output:** true **Explanation:** We can apply the following operations on s1: - Choose the indices i = 0, j = 2. The resulting string is s1 = "cbadba". - Choose the indices i = 2, j = 4. The resulting string is s1 = "cbbdaa". - Choose the indices i = 1, j = 5. The resulting string is s1 = "cabdab" = s2.

Example 2:

Input: s1 = "abe", s2 = "bea" **Output:** false **Explanation:** It is not possible to make the two strings equal.

Constraints:

* `n == s1.length == s2.length` * `1 <= n <= 105` * `s1` and `s2` consist only of lowercase English letters.

Code Snippets

C++:

```
class Solution {
public:
bool checkStrings(string s1, string s2) {

}
};
```

Java:

```
class Solution {
public boolean checkStrings(String s1, String s2) {

}
}
```

Python3:

```
class Solution:
def checkStrings(self, s1: str, s2: str) -> bool:
```