Problem 521: Longest Uncommon Subsequence I

Problem Information

Difficulty: Easy
Acceptance Rate: 61.78%
Paid Only: No
Tags: String

Problem Description

Given two strings `a` and `b`, return _the length of the**longest uncommon subsequence** between _`a` _and_ `b`. _If no such uncommon subsequence exists, return_ `-1` _._

An **uncommon subsequence** between two strings is a string that is a **subsequence of exactly one of them**.

Example 1:

Input: a = "aba", b = "cdc" **Output:** 3 **Explanation:** One longest uncommon subsequence is "aba" because "aba" is a subsequence of "aba" but not "cdc". Note that "cdc" is also a longest uncommon subsequence.

Example 2:

Input: a = "aaa", b = "bbb" **Output:** 3 **Explanation:** The longest uncommon subsequences are "aaa" and "bbb".

Example 3:

Input: a = "aaa", b = "aaa" **Output:** -1 **Explanation:** Every subsequence of string a is also a subsequence of string b. Similarly, every subsequence of string b is also a subsequence of string a. So the answer would be -1.

Constraints:

* `1 <= a.length, b.length <= 100` * `a` and `b` consist of lower-case English letters.

Code Snippets

C++:

```
class Solution {
public:
int findLUSlength(string a, string b) {

}
};
```

Java:

```
class Solution {
public int findLUSlength(String a, String b) {

}
}
```

Python3:

```
class Solution:
def findLUSlength(self, a: str, b: str) -> int:
```