521 Longest Uncommon Subsequence I (link)

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
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. Similar
```

Constraints:

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

(scroll down for solution)

Solution

Language: cpp

Status: Accepted

```cpp
#include <string>
#include <algorithm>

class Solution {
public:
    int findLUSlength(std::string a, std::string b) {
        if (a == b) {
            return -1; // Если строки идентичны, возвращаем -1
        } else {
            return std::max(a.length(), b.length()); // Возвращаем длину самой длинной
        }
    }
};
```