

Problem 3460: Longest Common Prefix After at Most One Removal

Problem Information

Difficulty: Medium

Acceptance Rate: 67.65%

Paid Only: Yes

Tags: Two Pointers, String

Problem Description

You are given two strings `s` and `t`.

Return the **length** of the **longest common prefix** between `s` and `t` after removing **at most** one character from `s`.

Note: `s` can be left without any removal.

Example 1:

Input: s = "madxa", t = "madam"

Output: 4

Explanation:

Removing `s[3]` from `s` results in `mada`, which has a longest common prefix of length 4 with `t`.

Example 2:

Input: s = "leetcode", t = "eetcode"

Output: 7

****Explanation:****

Removing `s[0]` from `s` results in `eetcode`, which matches `t`.

****Example 3:****

****Input:**** s = "one", t = "one"

****Output:**** 3

****Explanation:****

No removal is needed.

****Example 4:****

****Input:**** s = "a", t = "b"

****Output:**** 0

****Explanation:****

`s` and `t` cannot have a common prefix.

****Constraints:****

* `1 <= s.length <= 105` * `1 <= t.length <= 105` * `s` and `t` contain only lowercase English letters.

Code Snippets

C++:

```
class Solution {
public:
    int longestCommonPrefix(string s, string t) {
        }
};
```

Java:

```
class Solution {  
    public int longestCommonPrefix(String s, String t) {  
        }  
        }
```

Python3:

```
class Solution:  
    def longestCommonPrefix(self, s: str, t: str) -> int:
```