Problem 3628: Maximum Number of Subsequences After One Inserting

Problem Information

Difficulty: Medium
Acceptance Rate: 28.23%
Paid Only: No
Tags: String, Dynamic Programming, Greedy, Prefix Sum

Problem Description

You are given a string `s` consisting of uppercase English letters.

You are allowed to insert **at most one** uppercase English letter at **any** position (including the beginning or end) of the string.

Return the **maximum** number of `"LCT"` subsequences that can be formed in the resulting string after **at most one insertion**.

Example 1:

Input: s = "LMCT"

Output: 2

Explanation:

We can insert a `"L"` at the beginning of the string s to make `"LLMCT"`, which has 2 subsequences, at indices [0, 3, 4] and [1, 3, 4].

Example 2:

Input: s = "LCCT"

Output: 4

Explanation:

We can insert a `"L"` at the beginning of the string s to make `"LLCCT"`, which has 4 subsequences, at indices [0, 2, 4], [0, 3, 4], [1, 2, 4] and [1, 3, 4].

Example 3:

Input: s = "L"

Output: 0

Explanation:

Since it is not possible to obtain the subsequence `"LCT"` by inserting a single letter, the result is 0.

Constraints:

* `1 <= s.length <= 105` * `s` consists of uppercase English letters.

Code Snippets

C++:

```
class Solution {
public:
long long numOfSubsequences(string s) {

}
};
```

Java:

```
class Solution {
public long numOfSubsequences(String s) {

}
}
```

Python3:

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