Problem 1573: Number of Ways to Split a String

Problem Information

Difficulty: Medium
Acceptance Rate: 34.20%
Paid Only: No
Tags: Math, String

Problem Description

Given a binary string `s`, you can split `s` into 3 **non-empty** strings `s1`, `s2`, and `s3` where `s1 + s2 + s3 = s`.

Return the number of ways `s` can be split such that the number of ones is the same in `s1`, `s2`, and `s3`. Since the answer may be too large, return it **modulo** `109 + 7`.

Example 1:

Input: s = "10101" **Output:** 4 **Explanation:** There are four ways to split s in 3 parts where each part contain the same number of letters '1'. "1|010|1" "1|01|01" "10|10|1" "10|1|01"

Example 2:

Input: s = "1001" **Output:** 0

Example 3:

Input: s = "0000" **Output:** 3 **Explanation:** There are three ways to split s in 3 parts. "0|0|00" "0|00|0" "00|0|0"

Constraints:

* `3 <= s.length <= 105` * `s[i]` is either `'0'` or `'1'`.

Code Snippets

C++:

```
class Solution {
public:
int numWays(string s) {

}
};
```

Java:

```
class Solution {
public int numWays(String s) {

}
}
```

Python3:

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