Problem 2489: Number of Substrings With Fixed Ratio

Problem Information

Difficulty: Medium
Acceptance Rate: 57.06%
Paid Only: Yes
Tags: Hash Table, Math, String, Prefix Sum

Problem Description

You are given a binary string `s`, and two integers `num1` and `num2`. `num1` and `num2` are coprime numbers.

A **ratio substring** is a substring of s where the ratio between the number of `0`'s and the number of `1`'s in the substring is exactly `num1 : num2`.

* For example, if `num1 = 2` and `num2 = 3`, then `"01011"` and `"1110000111"` are ratio substrings, while `"11000"` is not.

Return _the number of**non-empty** ratio substrings of _`s`.

Note that:

* A **substring** is a contiguous sequence of characters within a string. * Two values `x` and `y` are **coprime** if `gcd(x, y) == 1` where `gcd(x, y)` is the greatest common divisor of `x` and `y`.

Example 1:

Input: s = "0110011", num1 = 1, num2 = 2 **Output:** 4 **Explanation:** There exist 4 non-empty ratio substrings. - The substring s[0..2]: "_011_ 0011". It contains one 0 and two 1's. The ratio is 1 : 2. - The substring s[1..4]: "0 _110_ 011". It contains one 0 and two 1's. The ratio is 1 : 2. - The substring s[4..6]: "0110 _011_ ". It contains one 0 and two 1's. The ratio is 1 : 2. - The substring s[1..6]: "0 _110011_ ". It contains two 0's and four 1's. The ratio is 2 : 4 == 1 : 2. It can be shown that there are no more ratio substrings.

Example 2:

Input: s = "10101", num1 = 3, num2 = 1 **Output:** 0 **Explanation:** There is no ratio substrings of s. We return 0.

Constraints:

* `1 <= s.length <= 105` * `1 <= num1, num2 <= s.length` * `num1` and `num2` are coprime integers.

Code Snippets

C++:

```
class Solution {
public:
long long fixedRatio(string s, int num1, int num2) {

}
};
```

Java:

```
class Solution {
public long fixedRatio(String s, int num1, int num2) {

}
}
```

Python3:

```
class Solution:
def fixedRatio(self, s: str, num1: int, num2: int) -> int:
```