# Problem 2450: Number of Distinct Binary Strings After Applying Operations

## Problem Information

**Difficulty:** Medium
**Acceptance Rate:** 63.47%
**Paid Only:** Yes
**Tags:** Math, String

## Problem Description

You are given a **binary** string `s` and a positive integer `k`.

You can apply the following operation on the string **any** number of times:

* Choose any substring of size `k` from `s` and **flip** all its characters, that is, turn all `1`'s into `0`'s, and all `0`'s into `1`'s.

Return _the number of**distinct** strings you can obtain_. Since the answer may be too large, return it **modulo** `109 + 7`.

**Note** that:

* A binary string is a string that consists **only** of the characters `0` and `1`. * A substring is a contiguous part of a string.

**Example 1:**

**Input:** s = "1001", k = 3 **Output:** 4 **Explanation:** We can obtain the following strings: - Applying no operation on the string gives s = "1001". - Applying one operation on the substring starting at index 0 gives s = "_**011**_ 1". - Applying one operation on the substring starting at index 1 gives s = "1 _**110**_ ". - Applying one operation on both the substrings starting at indices 0 and 1 gives s = "_**0000**_ ". It can be shown that we cannot obtain any other string, so the answer is 4.

**Example 2:**

**Input:** s = "10110", k = 5 **Output:** 2 **Explanation:** We can obtain the following strings: - Applying no operation on the string gives s = "10110". - Applying one operation on the whole string gives s = "01001". It can be shown that we cannot obtain any other string, so the answer is 2.

**Constraints:**

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

## Code Snippets

**C++:**

```
class Solution {
public:
int countDistinctStrings(string s, int k) {

}
};
```

**Java:**

```
class Solution {
public int countDistinctStrings(String s, int k) {

}
}
```

**Python3:**

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