1759. Count Number of Homogenous Substrings

Hint

Medium 966 61

Companies

Given a string `s`, return *the number of* ***homogenous*** *substrings of* `s`. Since the answer may be too large, return it **modulo** $10^9 + 7$.

A string is **homogenous** if all the characters of the string are the same.

A **substring** is a contiguous sequence of characters within a string.

Example 1:

```
Input: s = "abbcccaa"
Output: 13
Explanation: The homogenous substrings are listed as below:
"a"   appears 3 times.
```

Code

```
func countHomogenous(_ s: String) -> Int {
        var ans = 0
        var curr = Character("A")
        var currentCount = 0

        for i in s {
            if(curr == i) {
                currentCount = currentCount + 1
            } else {
                curr = i
                currentCount = 1
            }
            ans = ans + currentCount
        }

        return ans % 1_000_000_007
    }
```

```
// Solved wrong
    // func countHomogenous(_ s: String) -> Int {
    //     let modulo = 1_000_000_007
    //     if(s.count == 1) {return 1}
    //     var dict:[String:Int] = [:]
    //     let arr = Array(s)
    //     var temp:String = String(arr[0])
    //     for i in 1..<arr.count {
    //         // print(arr[i])
    //         if(arr[i-1] != arr[i]) {
    //             //  print(temp)
    //             dict[temp, default: 0] += 1
    //             temp = String(arr[i])
    //         } else {
    //             temp = temp + String(arr[i])
    //         }

    //         // print("****************")

    //     }
    //     dict[temp, default: 0] += 1
    //     // print(dict)

    //     var sumDict:[Int:Int] = [:]
    //     sumDict[1] = 1
    //     var count = 0
    //     for (i,v) in dict {
    //         if let y = sumDict[i.count] {
    //             // y = y * v
    //             count = (count + (y*v))
    //             //%(10^9 + 7)
    //         } else {
    //             let t = sumNum(i.count)
    //             count = (count + (t*v))
    //             //%(10^9 + 7)
    //             sumDict[t, default: 0] += 1
    //         }
```

```
//          // print(i)
//      }

//       //print(sumDict)

//      return count%modulo
// }

// func sumNum(_ n:Int) -> Int{

//      return (n*(n-1)/2)
// }
```