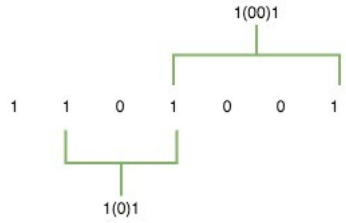
Pattern Count

by sarfraz1234

Your submission will run against only preliminary test cases. Full test cases will run at the end of the day.

A string s contains many patterns of the form $1(0+)1$ where $(0+)$ represents any non-empty consecutive sequence of 0's. The patterns are allowed to overlap.

For example, consider string **"1101001"**, we can see there are two consecutive sequences **"1(0)1"** and **"1(00)1"** which are of the form $1(0+)1$.



You have to answer q queries, each containing a string s. For each query, find and print the total number of patterns of the form $1(0+)1$ that occur in s.

Input Format

The first line contains a single integer q, denoting the number of queries. After that, q lines follow. The i^{th} of them represents the i^{th} test case and contains a string s for this test case.

Constraints

- $1 \leq q \leq 20$
- $1 \leq |s| \leq 2000$
- s contains only digits and lowercase English letters

Output Format

Output exactly q lines, one for each testcase.

Sample Input 0

```
3
100001abc101
1001ab010abc01001
1001010001
```

Sample Output 0

```
2
2
3
```

Explanation 0

- In the first case, $s =$ **"100001abc101"** we have $s[0 \cdots 5]$ as **"100001"** and $s[9 \cdots 11]$ as **"101"**. Hence, we print 2 as the answer.

- In the second case, $s =$ **"1001ab010abc01001"** we have $s[0 \cdots 3]$ as **"1001"** and $s[13 \cdots 16]$ as **"1001"**. Hence, we print **2** as the answer.
- In the third case, $s =$ **"1001010001"** we have $s[0 \cdots 3]$ as **"1001"**, $s[3 \cdots 5]$ as **"101"** and $s[5 \cdots 9]$ as **"10001"**. Hence, we print **3** as the answer.

Current Buffer (saved locally, editable) C#

```
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
class Solution {

    static int patternCount(string s){
        // Complete this function

         int cont = 0;
            bool hayuno = false;
            bool hayceroenmedio = false;
            for (int i = 0; i < s.Length; i++)
            {
                if (!hayuno)
                {
                    if (s[i] == '1')
                    {
                        hayuno = true;
                    }
                }
                else if (hayuno)
                {
                    if (s[i] == '0')
                    {
                        hayceroenmedio = true;
                    }
                    else if (s[i] != '0' && s[i] != '1')
                    {
                        hayuno = false;
                        hayceroenmedio = false;
                    }
                    else if (s[i] == '1')
                    {
                        if (hayceroenmedio)
                        {
                            cont++;
                        }
                        hayceroenmedio = false;
                    }
                }
                if (hayuno && hayceroenmedio)
                {
                    if (s[i] == '1')
                    {
                        cont++;
                    }
                }
            }

            return cont;
    }

    static void Main(String[] args) {
        int q = Convert.ToInt32(Console.ReadLine());
        for(int a0 = 0; a0 < q; a0++){
```

```
57              string s = Console.ReadLine();
58              int result = patternCount(s);
59              Console.WriteLine(result);
60          }
61      }
62  }
63
```

Line: 51 Col: 25

Upload Code as File Test against custom input Run Code Submit Code