# Now you are given a string containing characters A and B only. Your task is to help Krishna in changing it into a string such that there are no matching adjacent characters. To do this, you are allowed to delete zero or more characters in the string.

# Your task to help Krishna is to find the minimum number of required deletions.

# For example, given the string s=AABAAB, remove an A at positions 0 and 3 to make s=ABAB in 2 deletions.

**Input Format**

The first line contains an integer T, the number of queries.  
The next T lines each contain a string s.

**Constraints**

1<=q<=10

1<= |s| <= 105

Each string s will consist only of characters A and B

**Output Format**

For each query, print the minimum number of deletions required on a new line.

Java code is given below.

import java.util.\*;

class removeDuplicates {

static void removeDuplicates(char[] S)

{

int n = S.length;

if (n<2)

return ;

int c=0;

char ch=S[0];

for(int i=1;i<n;i++){

if(ch==S[i])

c++;

else

ch=S[i];

}

System.out.println(c);

}

public static void main(String[] args)

{

Scanner sc=new Scanner(System.in);

int t=sc.nextInt();

while(t-->-1){

String s=sc.nextLine();

char S1[] = s.toCharArray();

removeDuplicates(S1);

}

}

}