**Encrypt the string - 1**

Show Topic Tags   

Bingu was testing all the strings he had at his place, and found that most of them were prone to vicious attack by Banju, his arch enemy. Bingu decided to encrypt all the strings he had, by the following method. Every substring of identical letters is replaced by a single instance of that letter followed by the number of occurences of that letter. Then, the string thus obtained is further encrypted by reversing it.

**Input:**  
The first line contains T, the number of test cases. Then T lines follow, each containing a string S ( |S| <= 1000), consisting of lower case English alphabets  
  
**Output:**  
For each test case in a new line print the required encrypted string.

**Constraints**  
1<=T<=28  
Length of each string <=1000  
  
  
**Example:  
Input:**  
2  
aabc  
aaaaa  
  
**Output:**  
1c1b2a  
5a  
  
**Explanation:**  
**For first test case**  
aabc  
Step1: a2b1c1  
Step2: 1c1b2a

\*\*For More Examples Use Expected Output\*\*

<http://practice.geeksforgeeks.org/problems/encrypt-the-string-1/0>

/\*

\* To change this template, choose Tools | Templates

\* and open the template in the editor.

\*/

package javaapplication245;

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStreamReader;

import java.util.ArrayList;

import java.util.Arrays;

/\*\*

\*

\* @author Administrador

\*/

public class JavaApplication245 {

/\*\*

\* @param args the command line arguments

\*/

public static void main(String[] args) throws IOException {

// TODO code application logic here

BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

int t = Integer.parseInt(br.readLine());

while(t-- > 0) {

String s = br.readLine().trim();

int i =s.length()-1;

String concat = "";

int cont = 1;

char actual = ' ';

while(i >=0 ) {

actual =s.charAt(i);

cont = 0;

while(i >=0 && s.charAt(i) ==actual) {

i--;

cont++;

}

StringBuffer sb = new StringBuffer(String.valueOf(cont)).reverse();

concat += sb.toString() + actual;

}

System.out.println(concat);

}

}

}