UVa Online Judge

1261 String Popping

We are given a string s of two characters ‘`a`’ and ‘`b`’. Let a *group* be a maximal consecutive substring of the same character. Any group g of s of length at least two can be removed (or *popped*) and a new string is constructed by concatenating the remaining left and right substrings of s. We repeat this process until either the string becomes the empty string or there is no more group of length at least two.

For example, string $s = babbbaaabb$ has 5 groups b, a, bbb, aaa, and bb. The string s can be turned into the empty string by popping groups in the following sequence (the underlined group is to be popped in the sequence):

$$ba\underline{bbb}aaabb \rightarrow b\underline{aaaa}bb \rightarrow \underline{bbb} \rightarrow \textit{empty string}$$

But the group may not turn to an empty string by a different sequence of pop operations:

$$babbbaaa\underline{bb} \rightarrow ba\underline{bbb}aaa \rightarrow b\underline{aaaa} \rightarrow b$$

Given a string, write a program to decide whether the string can be turned into the empty string by some sequence of popping operations.

Input

Your program is to read from standard input. The input consists of T test cases. The number of test cases T is given in the first line of the input. Each test case consists of a single line containing a string of two characters ‘`a`’ and ‘`b`’. The minimum and maximum length of the string is 1 and 25, respectively.

Output

Your program is to write to standard output. Print one line for each test case. The line of each test case should contain ‘`0`’ or `1`’. Print `1`’ if the input string can be turned to an empty string by a sequence of popping operations. Otherwise print ‘`0`’.

The following shows sample input and output for three test cases.

Sample Input

```
3
babbbaaabb
aabbaabb
abab
```

Sample Output

```
1
1
0
```