String Reduction

Given a string consisting of letters, 'a', 'b' and 'c', we can perform the following operation:

- Take any two adjacent distinct characters and replace them with the third character.

For example, if 'a' and 'c' are adjacent, they can replaced by 'b'.

Find the smallest string which we can obtain by applying this operation repeatedly.

Input Format

The first line contains the number of test cases T. T test cases follow. Each test case contains the string you start with.

Constraints

- $1 \leq T \leq 100$
- The string will have at most 100 characters.

Output Format

Output T lines, one for each test case, containing the smallest length of the resultant string after applying the operations optimally.

Sample Input

```
3
cab
bcab
ccccc
```

Sample Output

```
2
1
5
```

Explanation

For the first case, you can either get $cab -> cc$ or $cab -> bb$, resulting in a string of length 2.
For the second case, one optimal solution is: $bcab -> aab -> ac -> b$. No more operations can be applied and the resultant string has length 1.
For the third case, no operations can be performed. So the answer is 5.