Problem C. Making Huge Palindromes

Time limit 1000 ms
Mem limit 65536 kB

A string is said to be a palindrome if it remains same when read backwards. So, "abba", "madam" both are palindromes, but "adam" is not.

Now, you are given a non-empty string **S**, containing only lowercase English letters. The given string may or may not be palindrome. Your task is to make it a palindrome. But you are only allowed to add characters at the right side of the string. And of course, you can add any character you want, but the resulting string has to be a palindrome, and the length of the palindrome should be as small as possible.

For example, the string is **"bababa"**, you can make many palindromes including:

- "bababa**babab**"
- "bababa**bab**"
- "bababa**b**"

Since we want a palindrome with minimum length, the solution would be **"bababab"** cause its length is minimum.

Input

Input starts with an integer **T (≤ 10)**, denoting the number of test cases.

Each case starts with a line containing a string **S**. You can assume that **1 ≤ length(S) ≤ 10^6**.

Output

For each case, print the case number and the length of the shortest palindrome you can make with **S**.

Sample

Input	Output
4 bababababa pqrs madamimadam anncbaaababaaa	Case 1: 11 Case 2: 7 Case 3: 11 Case 4: 19

Note

Dataset is huge, use faster I/O methods.