

Day 6: Love Letter Mystery

James found a love letter his friend Harry wrote to his girlfriend. Being a prankster, he decides to make some fun adjustments to it by changing each of its words into [palindromes](#).

To do this, he follows two rules:

1. He can *reduce* the value of a letter (e.g.: he can reduce *d* to *c*, but can't increment *c* to *d*).
2. A letter can be reduced more than once, but once it is reduced to *a* it becomes locked at *a* and its value can no longer be changed.

Each reduction in a letter's value is counted as \$1\$ operation. For each test case, find the *minimum* number of operations required to convert the given string to a palindrome.

Input Format

The first line contains an integer, \$T\$ (the number of test cases).
Each of the \$T\$ subsequent lines contains a single string of lowercase characters.

Constraints

\$1 \le T \le 10\$
\$1 \le \text{length of string} \le 10^4\$
All characters are lowercase English letters.

Output Format

For each test case, print a new line with the minimum number (as an integer) of operations required to convert the given string to a palindrome.

Sample Input

```
4
abc
abcba
abcd
cba
```

Sample Output

```
2
0
4
2
```

Explanation

Test Case 0: **ab**c \rightarrow **ab**b \rightarrow **aba**. We print the number of operations, \$2\$, on a new line.

Test Case 1: **abcba** is already a palindrome. We print the number of operations, \$0\$, on a new line.

Test Case 2: **abc**d \rightarrow **abc**c \rightarrow **abc**b = **abca**. Then **ab**c**a** \rightarrow **abba**. We print the number of operations, \$4\$, on a new line.

Test Case 3: **c** **ba** \rightarrow **b** **ba** \rightarrow **aba**. We print the number of operations, \$2\$, on a new line.