

A **palindrome** is a string that reads the same from the left as it does from the right. Given two strings A and B , you need to find the length of longest palindrome which is a subsequence of both A and B . A subsequence is a sequence obtained by deleting zero or more characters from a string.

For example, say, $A = \text{"cfcfaafc"}$, $B = \text{"efagfc"}$. Then the longest palindrome which is a subsequence of both A and B is **"faf"**. So the answer is 3.

Input

First line of the input contains a positive integer T ($T \leq 100$). Each of the following T cases consists of 2 lines. These 2 lines contain the strings A and B , respectively. Length of A and B will not be more than 60. All these strings contain only lowercase letters ('a' - 'z'). No empty strings will appear in the input.

Output

For each case, print a line of the form **'Case x: y'**, where x is the case number and y is the length of the longest common palindromic subsequence.

Sample Input

```
3
cfcfaafc
efagfc
afbdfca
bcadfcgyfka
palin
drome
```

Sample Output

```
Case 1: 3
Case 2: 5
Case 3: 0
```