1351 – Ordered Flips

You are given two non-empty strings **X** and **Y** of same length **n**. Your task is to make them identical. But the problem is that the only operation you can do is flipping, and it can only be applied to **X**.

For a flip, two positions of the string **X** are chosen, let the positions be **i** and **j (0 ≤ i < j < n)** and if you apply **flip (i, j)** all characters between **i** and **j** (inclusive) are reversed. For example, let **X** be **"abcdefg"**, then if you apply flip (2, 5) to **X** then **X** will be **"abfedcg"**. But if you want to apply flips more than once, you have to use ordered flips. If **flip (i_2, j_2)** is applied immediately after **flip (i_1, j_1)**, then it will be said **"Ordered Flips"** if and only if $i_1 \le i_2$ and $j_2 \le j_1$.

So, now your task is to find the minimum number of ordered flips to change **X** to **Y**.

Input

Input starts with an integer **T (≤ 200)**, denoting the number of test cases.

Each case contains two lines, each containing a non empty string of length **n (1 ≤ n ≤ 60)**. The strings contain lowercase English letters only. First line contains **X** and second line contains **Y**.

Output

For each case, print the case number and the minimum number of ordered flips needed to change **X** to **Y**. If it's impossible to do, then print **"impossible"**. Check the samples for details.

Sample Input	Output for Sample Input
4 abcd dcba abca aabc zzzaaazzzaaa aazzaazzaazz aab bab	Case 1: 1 Case 2: 2 Case 3: 4 Case 4: impossible