

Rearranging Scrabble Tiles

Filename: tiles

Time Limit: 1 second

You've been playing a lot of Scrabble lately. When you change the order of the tiles, you only swap consecutive tiles. You're curious, given the current state of your tiles and a desired word to form (that uses the exact same letters), what is the minimum number of tile swaps that will be necessary to get the tiles into their desired order?

Problem

Given the original order of a set of letter tiles, and a desired order to change the tiles into, given that you can only swap consecutive tiles, calculate the minimum number of swaps necessary to get the tiles into their desired order.

Input

The first line will contain a single integer c ($1 \leq c \leq 10$), the number of test cases. The input cases follow, one per line. Each input case will be two space separated strings, s and t ($1 \leq |s| = |t| \leq 26$), where s is the desired ordering of the Scrabble tiles and t represents the original ordering of the tiles. It is guaranteed that s has distinct uppercase letters and that t is a permutation of the letters in s .

Output

For each case, on a line by itself, output the minimum number of swaps necessary to modify the rack of tiles to the desired order.

Sample Input/Output

Input	Output
3	9
CARPET TPACRE	13
TRAVIS SIRVAT	2
CAB ABC	