Bits Equalizer

Hide

You are given two non-empty strings S and T of equal lengths. S contains the characters '0', '1' and '?', whereas T contains '0' and '1' only. Your task is to convert S into T in minimum number of moves. In each move, you can

1. change a '0' in S to '1',
2. change a '?' in S to '0' or '1', or
3. swap any two characters in S.

As an example, suppose $S =$ "01??00" and $T =$ "001010". We can transform S into T in 3 moves:

- Initially $S =$ "01??00"
- Move 1: change $S[2]$ to '1'. S becomes "011?00".
- Move 2: change $S[3]$ to '0'. S becomes "011000"
- Move 3: swap $S[1]$ with $S[4]$. S becomes "001010"
- S is now equal to T.

Input

The first line of input is an integer C ($C \leq 200$) that indicates the number of test cases.

Each case consists of two lines. The first line is the string S consisting of '0', '1' and '?'. The second line is the string T consisting of '0' and '1'. The lengths of the strings won't be larger than 100.

Output

For each case, output the case number first followed by the minimum number of moves required to convert S into T. If the transition is impossible, output -1 instead.

Sample Input 1

```
3
01??00
001010
```

Sample Output 1

```
Case 1: 3
Case 2: 1
Case 3: -1
```

```
01
10
110001
000000
```

Hide >

Please log in to submit a solution to this problem

Log in