

# Challenge Accepted

Aman and Arpit are bored of coding. There is no challenge in it anymore, they feel. Bear in mind that even the problems which would appear very difficult for most of you are trivial for them. With ICPC approaching, their teammate, Sameer, is worried by this development. So, to rekindle their interest in coding, he challenges them to a game.

The game consists of an  $N \times M$  grid ( $N$  rows,  $M$  columns). Some of the cells in the grid are occupied by coins. You can assume that all coins are identical. Sameer then gives them another grid of the same size, with a possibly different arrangement of coins. The objective is simple: the initial grid has to be transformed into the final grid, by moving coins. The following rules apply for coin movements:

1. In one move, exactly one coin can be moved.
2. A coin can move one cell up, down, left or right in one move, provided that the destination cell is empty.
3. If the destination cell is occupied by another coin, then the current coin can jump over that particular coin to land in an empty cell. In fact, a coin can jump over any number of consecutive coins until it lands in an empty cell. It cannot jump over an empty cell. Irrespective of the number of coins it jumps over, it is still counted as one move.

Whoever solves this in minimum number of moves will be the winner and will get a treat from Sameer. Since both Aman and Arpit really want to win, they immediately proceed to write code for the same. Can you help them predict the minimum number of moves in which the initial grid can be transformed into the final one?

## Input

First line of input contains  $T$ , the number of test cases.

First line of each test case contains 2 integers,  $N$  and  $M$ , separated by a single space.

This is followed by  $2 \times N$  lines per test case. The first  $N$  lines represent the initial grid and the next  $N$  lines represent the final grid.

Each character in the grid is either a 'C' or a '.'

'C' means that there is a coin on that cell, and '.' means that the cell is empty.

## Output

For each test case, output a single line containing the minimum number of moves required. If it is not possible to transform the original grid into the final one, output '-1' (without quotes)

## Example

Input:

2

1 5

.CC..

CC...

3 3

C.C

.C.

C.C

...

CCC

.CC

**Output:**

1

3

Explanation: In the first test case, the coin to the right can jump over the coin to the left in 1 move.

**Constraints:**

$1 \leq T \leq 10$

$1 \leq N * M \leq 20$