J. Edit distance on table

Problem ID: J

You have a table with H rows and W columns. Each cell of the table contains a letter. You are going to construct a string by the following steps.

- Step 1: Pick up a cell in the table and let S be a string of length 1 containing the letter in the cell.
- Step 2: Do either
 - stop building S, or
 - select a cell from four cells which shares an edge with the current one. Then, append the letter in the cell to S, and move to the cell. Then, repeat step 2.

You also have a string T. Your mission is to minimize the edit distance between S and T.

The edit distance (also known as Levenshtein distance) between string U and V is the minimum number of steps required to convert U into V by using the following operations.

- Replace a character in U with another one.
- Insert a character into U.
- Delete a character from U.

Input

The input consists of a single test case in the following format.

```
H W \\ c_{1,1}c_{1,2} \dots c_{1,W} \\ c_{2,1}c_{2,2} \dots c_{2,W} \\ \vdots \\ c_{H,1}c_{H,2} \dots c_{H,W} \\ T
```

H and W represents the height and the width of the table respectively. $c_{i,j}$ is a character in the cell in the *i*-th row and the *j*-th column. T is a non-empty string.

- $2 \le H, W \le 100$
- $|T| \le 2000$
- $c_{i,j}$ and T consist of lowercase English letters

Output

Output the minimum possible edit distance between S and T in one line.

Sample Input 1	Sample Output 1
2 2	2
ab	
ar	
abracadabra	