international collegiate programming contest
ASIA REGIONAL CONTEST
ICPC JAKARTA 2022

BINUS
UNIVERSITY

icpc.foundation

Practice Problem PD

Cell Game

Two brothers, Aldo and Bondan, are stuck in their home as their city is going into lockdown again due to the worsening situation of the COVID-19 pandemic. They have finished their semester and are on holiday, but what kind of holiday can you enjoy if you cannot get out of your house. However, boredom does spark creativity. They created a new game during their boring holiday.

The game is played on a board of R rows and C columns where each cell contains zero or one token. Each token is painted with one of the 26 colors that is represented by a character from 'a' to 'z'. There is at least one token on the board.

Two opposing players play alternatingly. In their turn, the player chooses one token and move it to a not necessarily empty adjacent cell (i.e. in north, south, west, or east direction). A move is a **good move** if and only if the player moves a token of color x to a cell that already contains a token of the same color x. The objective of the game is to have as many good moves as possible. The player with strictly more good moves wins the game. If both players have the same number of good moves, then it's a tie and no one wins.

This game can be played indefinitely. However, Aldo and Bondan agree to play it for a total of 10^{100} moves for both with Aldo having the first move.

Bondan thinks that this kind of game is boring, so he decides to play it lazily. Whenever it is Bondan's turn, he will choose the same token that has just been moved by Aldo in his last previous turn, and move that token to an adjacent cell uniformly at random.

Despite that, Bondan doesn't like to lose. Before the game starts, he might need to alter the board by changing the board size or moving the tokens' initial location such that there is exactly zero chance for Aldo to win the game even though Bondan plays lazily. Specifically, the board can be extended from $R \times C$ into $R' \times C'$ where $R' \geq R$ and $C' \geq C$, and each token's initial location can be moved to another cell while ensuring that each cell contains at most one token. No token can be discarded and no new token can be added to the board.

Your task in this problem is to find a new board setting such that there is zero chance for Aldo (the first player) to win the game with a total of 10^{100} played moves although Bondan plays it lazily. The new board setting should have a minimum total number of cells. If there is more than one solution, you only need to output (any) one of them.

Input

Input begins with a line containing two integers R C ($1 \leq R, C \leq 1000$) representing the initial number of rows and the number of columns in the board, respectively. It is guaranteed that there are at least 2 cells on the board. Each of the next R lines contains C characters representing the initial board condition. The character '.' represents an empty cell while a character from the lowercase alphabets ('a-z') represents a token with such a color. It is guaranteed that there is at least 1 token on the board.

Output

Output begins with a line containing two integers R' C' representing the number of rows and the number of columns in the new board, respectively. Each of the next R' lines contains C' characters representing the new board condition. The character '.' represents an empty cell while a character from the lowercase alphabets ('a-z') represents a token with such a color. The new board should guarantee a zero chance for Aldo (the first player) to win the game with a total of 10^{100} played moves although Bondan plays it lazily. It should have a minimum total number of cells and should have the same set of tokens as the initial board.

Sample Input #1

```
2 3
ab.
c.d
```

Sample Output #1

```
2 3
ab.
c.d
```

Explanation for the sample input/output #1

There are no two tokens with the same color, thus, it is not possible for any player to have a good move. Aldo cannot win this game.

Sample Input #2

```
2 3
aa.
c.d
```

Sample Output #2

```
2 3
a.a
c.d
```

Sample Input #3

```
1 2
oo
```

Sample Output #3

```
1 3
o.o
```