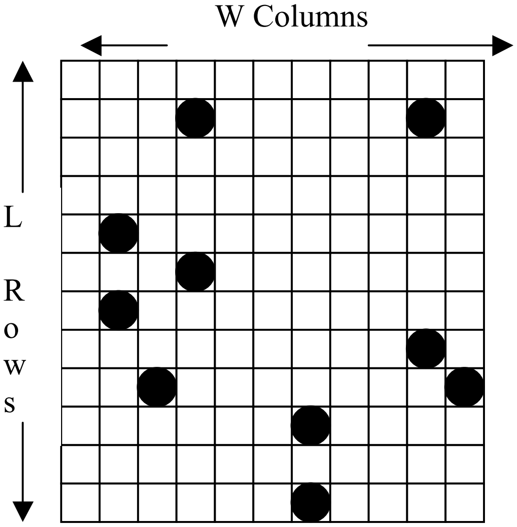
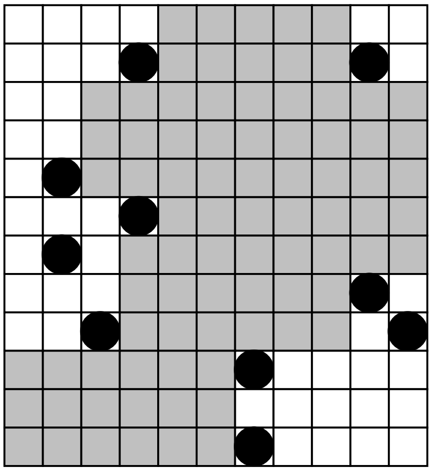
E - Mowing
Input File: MowingIn.txt

Uncle Mo mows rectangular lawns that are L feet long and W feet wide. The lawns are divided into 1 foot by 1 foot squares, and some of these squares have trees growing within them (shown as black circles in the figure below).



Uncut Lawn



Cut Lawn (shaded portion)

Mo has a 3 x 3 foot square lawn mower that can only move parallel to the edges of the lawn (vertical or horizontal). It must remain entirely within the rectangular lawn at all times and can never overlap a square containing a tree. The lawn mower cannot be rotated and cannot be picked up and moved over a tree.

You are to write a program to determine which squares of the lawn can be cut ("reached") by the lawn mower (the shaded region in the right portion of the above figure assuming the mower' intial position was row 5, column 6).

Inputs

The first line of the input contains the number of lawns to be cut. This is followed by one data set for each lawn. The first line of a data set contains four positive integers, the length and width, L and W, followed by row and column number of the initial position of the center of the lawn mower (row 0, column 0 is the upper left square of the lawn). Each of the next L lines contains W marker characters, each of which corresponds to a square on the lawn. The marker character **T** means that there is a tree on the square and the marker character **.** means no tree. To aid readability, each marker character is preceded by a single space. (You may assume that length and width are at most 40.)

Outputs

For each lawn, there should be L lines of output. Each output line will contain W marker characters, each preceded by a single space. This grid of marker characters should be the same as the input lawn description except that the **.** markers shouild be replaced with a C marker character wherever a square of the lawn can be cut by the mower. The output lawns should be separated by one blank line.

Sample Input

```
2
12 11 5 6
. . . . . . . . . . .
. . . T . . . . . T .
. . . . . . . . . . .
. . . . . . . . . . .
. T . . . . . . . . .
. . . T . . . . . . .
. T . . . . . . . . .
. . . . . . . . . T .
. . T . . . . . . . T
. . . . . . T . . . .
. . . . . . . . . . .
. . . . . . T . . . .
11 13 9 1
. . . T . . . . . . . . T
. . . . T . . . . . . . .
. . . . . . . . . . . . .
. . . . . . . . T . . . .
. . . . T . . T . . . . .
T . . . . . . . . . . . .
. . . . . . . . . . . . .
. . . . . . . . . . . . T
. . . . . . . . T . . . .
. . . . . . . T . . . . .
. . . T . . T . . . . . .
```

Sample Output

```
. . . . C C C C C . .
. . . T C C C C C T .
. . C C C C C C C C C
. . C C C C C C C C C
. T C C C C C C C C C
. . . T C C C C C C C
. T . C C C C C C C C
. . . C C C C C C T .
. . T C C C C C C . T
C C C C C C T . . . .
C C C C C C . . . . .
C C C C C C T . . . .

C C C T . C C C C C C C T
C C C C T C C C C C C C C
C C C C . C C C C C C C C
C C C C . C C C T C C C C
C C C C T . . T C C C C C
T C C C C C C C C C C C C
C C C C C C C C C C C C C
C C C C C C C C C C C C T
C C C C C C C C T C C C C
C C C C C C C T . C C C C
C C C T . . T . . C C C C
```