Terraforming Drones

Difficulty: Medium

Tagline: An alien species deploys Terraforming Drones on a rectangular grid, initially placing some drones randomly. After a brief inactive phase, new drones are placed to fill all empty cells and periodically initiate bursts that destroy itself and the 4 neighbouring drone (in front, behind, left and right). This burst will also transform the the section of land allocated to each drone on the grid. The challenge is to determine the state of the grid after several seconds of drone activity.

Problem Statement:

An alien species has deployed a fleet of "Terraforming Drones" onto a rectangular grid of land on Earth. Each cell in the grid represents a section of land, and each cell either contains a drone or is currently empty.

The Terraforming Drones operate in a specific way to reshape the environment:

1. Initially, some of the drones are placed randomly on the grid in the "Initial Deployment" phase.
2. After one second, the drones enter an "Inactive" phase, where they just hover in place, doing nothing.
3. After one more second, new Terraforming Drones are deployed in every empty section of land. This means that all empty cells are now occupied by drones, effectively covering the entire grid.
4. After another second, any drones deployed exactly three seconds ago initiate their "Terraforming Burst." When a drone bursts, it releases a pulse that destroys itself and transforms the land it has been assigned as well and the land in the four neighboring cells (front, back, left and right). If there are other drones in those neighboring cells (front, back, left and right), they are also destroyed before they can burst, resulting in no chain reaction.

The drones continue to repeat steps 3 and 4 as time passes.

Given the initial deployment configuration of drones on the grid, determine the state of the grid after n seconds.

For example, if the initial grid looks like:

. . .

. D .

. . .

After the first second, the grid remains the same, with the drone hovering in place.

After the second second, new Terraforming Drones are deployed to all empty sections:

DDD

DDD

DDD

At the third second, the original drone bursts, transforming the land around it and destroying itself:

D . D

. . .

D . D

Complete the alienTerraforming function below:

alienTerraforming has the following parameter(s):

- int n: the number of seconds to simulate
- string grid[r]: an array of strings that represents the grid

Input Format:

- The first line contains three space-separated integers r, c, and n, The number of rows, columns and seconds to simulate.
- Each of the next r lines contains a row of the matrix as a single string of c characters.
- The **.** character denotes an empty cell, and the **D** character denotes a drone.

Constraints:

- $1 \leq r,c \leq 200$
- $1 \leq n \leq 10^9$

Output Format:

- string[r]: an array of strings that represent the grid in its final state.

Sample Input:

STDIN Function
----- --------
6 7 3 r = 6, c = 7, n = 3
..... grid = ['.....', '...D...', '....D..',\
...D... '.....', 'DD.....', 'DD.....']
....D..
.....
DD.....
DD.....

Sample Output:

DDD.DDD

DD...DD

DDD...D

..DD.DD

...DDDD

...DDDD

Explanation:

The initial state of the grid is:

.....

...D...

....D..

.....

DD.....

DD.....

Drones spend the first second doing nothing, so this is the state after 1 second:

.....

...D...

....D..

.....

DD.....

DD.....

New drones are deployed in all the empty cells during the second second, so this is the state after 2 seconds:

DDDDDDD

DDDDDDD

DDDDDDD

DDDDDDD

DDDDDDD

DDDDDDD

In the third second, Drones start to burst. This is the final state after 3 seconds:

DDD.DDD

DD...DD

DDD...D

..DD.DD

...DDDD

...DDDD