Bubbles

In the long and not really interesting history lessons Elly loves playing different games on her smartphone. In the moment her passion is the game Bubbles. It has the following rules.
Colored bubbles are arranged in **N** rows, each consisting of **M** columns. The game has **K** rounds. In each round the player chooses a row and a column and pops the balloon there (if there is one). Together with it pop also all balloons with the same color neighboring to it, then the ones neighboring to them and so on, until there is no other neighboring balloon of the same color to a currently popped one. After this process ends (which you can consider to happen instantaneously) possibly some of the non-popped balloons may stay in the air. These balloons start falling until either reaching another non-popped balloon or the ground.
For example let's see what happens if we have the following configuration with 5 rows and 20 columns. With different capital letters we've denoted balloons of different colors. The player chooses to pop the bolded one with color 'A':

```
AAAEECCCCBBBAAEEPTTR    AAAEECCCCBBBAAEEPTTR    AAAEE         EEPTTR
AAAAECAACAABBAAAKAAA    AAAAEC  CAABBAAAKAAA    AAAAE      BAAAAKAAA
ACCCCAAESPRITBBBBAAA    ACCCC  ESPRITBBBBAAA    ACCCC  CCBBBBABBBAAA
AZCCCAAF AAABBBASSSAA    AZCCC  F   BBBASSSAA    AZCCCC ECAAITBASSSAA
ACCCCAAAAAAAAAVAAPAA    ACCCC         VAAPAA    ACCCCCCFSPRBBBVAAPAA
```

The rules of the game state that if the player has popped P balloons in this round, he gets P*P points. Thus, the game encourages choosing (and creating!) larger groups of balloons of the same color. The final score is the sum of the points for among all rounds.

Elly wants to impress Stancho, who is quite good at this game. Stancho has already played, so she knows the initial configuration of the balloons, the number of rounds, and the score he got. Help her by writing a program, that finds a sequence of pops that would get Elly as greater final score as possible.

Input
On the first line of the input file **`Bubbles.in`** will be given the integer **T** – how many games will Elly and Stancho play. **T** tests follow, separated by one blank line. Each test begins with a line, containing three integers N_i, M_i and K_i – the number of rows and columns, as well as the number of pops Elly must do. N_i lines follow, each containing M_i capital characters of the English alphabet ({'A'-'Z'}). Please note that the first line is the *topmost* row of balloons! On the last line for this test stays an integer S_i – the score that Stancho got for this game. You will be graded relatively to this score (see "Grading" for details).

Output
In this problem you must send us the output only, not the solution with which you got it.
In a file with name **`Bubbles.out`** print sequence of cells, the balloons in which must be popped. For each game (test) print K_i rows, each containing two integers – the row and the column of the popped balloon. The indexing starts from 0 and (0, 0) is the cell in the lower left corner of the table.
Note that sometimes it makes sense to pop the same cell more than once in the same game, since the balloons are being re-arranged between the rounds. It is also allowed to "pop" a cell, that does not contain a balloon, which doesn't change anything and doesn't bring you

points. Such a move would be appropriate, if for example you've already popped all balloons.

Constraints

- $1 \le N_i, M_i, K_i \le 50$
- $1 \le S_i \le N_i * N_i * M_i * M_i$

Sample Input	*Sample Output*
2 5 20 4 AAAEECCCCBBBAAEEPTTR AAAAECAACAABBAAAKAAA ACCCCAAESPRITBBBBAAA AZCCCAAFAAABBBASSSAA ACCCCAAAAAAAAAVAAPAA 1284 3 23 11 AAAAAAAAAAAAAAAAAAAAAAA BCBCBCBCBCBCBCBCBCBCBCB AAAAAAAAAAAAAAAAAAAAAAA 1229	1 8 0 0 0 1 3 18 1 0 1 1 1 2 1 3 1 4 1 5 1 6 1 7 1 8 2 22 0 0

*In the first round of the first game, Elly chooses a cell that leads to popping 18 balloons with color 'A' (around the center of the table). This is exactly the move, given as example earlier in the statement. It gives her 18*18 = 324 points. In the second round she pops another group of balloons with color 'A', but this time in the left part of the table. The group contains 10 balloons and gives her 100 points. The third round annihilates 14 balloons with color 'C' and brings Elly 196 points. The last round deals with a third group of balloons with color 'A' in the right part of the table, bringing again 100 points. In total the girl has gained 324 + 100 + 196 + 100 = 720 points. Since Stancho has won 1284 points, this solution will get round(720 / 1284 * 10) = round(5.60747664) = 6 points for this game.*

In the second game Elly has gained exactly as many points as Stancho, thus she gets 10 points for it.

Grading

For each game, if your solution has gained Y points after all rounds you will get round(min(1, Y / S_i) * 10) points – thus proportional to Stancho's score, rounded to the nearest integer in the range [0, 10].

Visualizer

A visualizer has been provided for your convenience. It tests the validity of your output, as well as calculates the real points you will get with a certain solution. It also displays what moves your solution does and how the board changes after each round.

The visualizer expects the input in a file with name **`Bubbles.in`** and the sequence of moves in a file with name **`Bubbles.out`**. Both files must be in the same directory as the executable of the visualizer. You can easily change the code so it stops printing the board after each round or how much time (in milliseconds) to wait between each print.