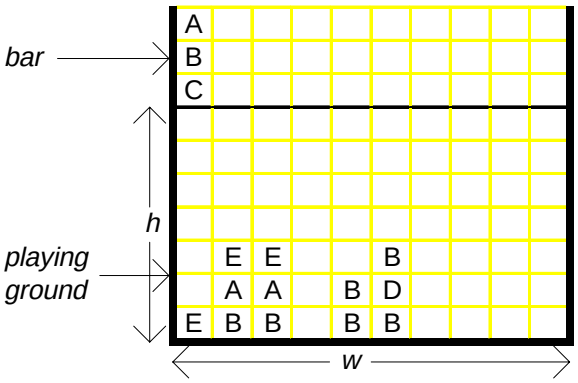
Problem D : Columns

inputfile: input.D
outputfile: output.D

Description

The game *Columns* is rather well known, at least in its dynamic form. The player can manipulate a bar of coloured squares while it drops down. Once it has reached the playing field, points are gained and squares in the playing field disappear, according to certain rules. This game is very exciting, as the bars will drop down faster and faster as the game goes on. For some people it turns out to be too exciting, as they get a nervous breakdown from playing this game. For those people the static variant was invented.
(Of course not playing at all would be a better idea. But the real addicts get a nervous breakdown if they don't play as well.)

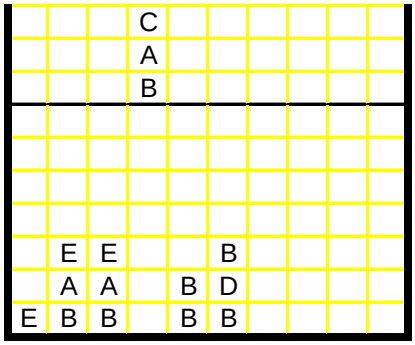
Let me explain *Static Columns* to you. The playing field is a rectangle of width w and height h. Initially it is empty, but after some time it may be filled with coloured squares. In the scheme below the colours are indicated with capitals. The squares are subject to gravity: a square rests either on the floor or on another square. Repeatedly a vertical bar (width 1) of coloured squares appears on top of the playing field, against the left wall.



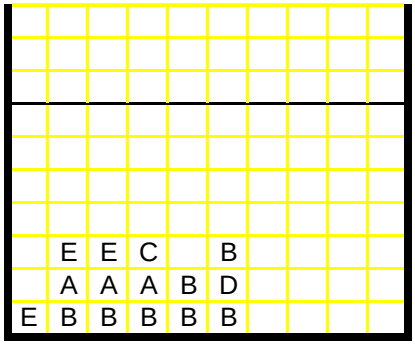
The player may manipulate this bar using (repeatedly) any of the following commands:

Command	Action
r (right)	Move the bar one step to the right. This command has no effect if the bar is already at the rightmost position of the playing field.
l (left)	Move the bar one step to the left. This command has no effect if the bar is already at the leftmost position of the playing field.
s (shift)	Shift (or actually rotate) the colours in the bar. The rotation is downwards: every colour goes down one step, and the lowermost colour is put on top.
d (drop)	Drop the bar. After this command, no other commands can be given for that bar.

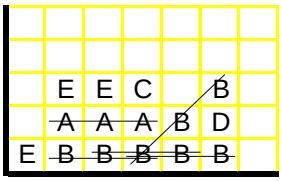
After the commands `rrrs` the situation in the above example will be:



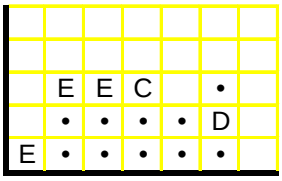
The drop command makes the bar going straight down until it is stopped, either by the floor or by a coloured square.



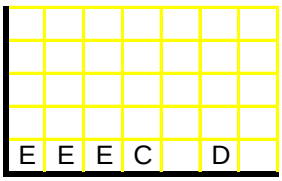
If the playing ground is very full, or the bar is very long it may happen that the bar is stopped before it is fully inside the playing ground. In that case the game is over and no other bars will be played. If not, two things happen. First, credits are counted, depending of the parameter l, the length of a series. Whenever l squares in the field in a series (horizontal, vertical or diagonal) have the same colour, one point is given to the player. As indicated in the next diagram, the player receives 5 points if $l = 3$ (the player would receive 2 points if $l = 4$ and 1 point if $l = 5$).



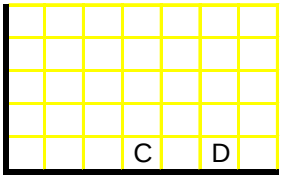
After all squares occurring in one or more such a series have disappeared, the following diagram is obtained:



Now gravity takes over, and the diagram becomes:



Once again a series is found and the player obtains one more point. The final situation is:



Problem

The *World Championships Static Columns* will be held next year in a very obscure place slightly North of nowhere. Because this place is far away from everywhere, not many people will actually go there, but lots of people are anxious to know what is going on there. Using the Internet every move of every player is broadcasted all over the world. People with beautiful graphics on their machine will be able to replay the game as if they were actually there. Others are less well supplied, and are happy if they know the credits obtained by each player. You are asked to write a program to calculate the credits.

Input

The first line of input contains the number of games g. Then g games follow.

Every game starts with a line with 4 numbers, the parameters of the game. The first number is the width w of the board: $(1 \leq w \leq 500)$. The next number is the height h of the board: $(1 \leq h \leq 250)$, The third number is the length l of a series: $(1 < l < 10)$. The last number is the number of moves m to follow. Then m moves are given, each on a line.

A move is given by a sequence of characters. It starts with one or more capitals, indicating the colours of the bar, from top to bottom, followed by the commands of the player, represented by the letters `l` (left), `r` (right) and `s` (shift). A move always ends with the letter `d` (drop). A move is never longer than *5000* characters.

Output

The output consists of g lines, one line for each game. In case of game over this line consists of the text `GAME OVER`, followed by a space and the credits gained until the game was over. (Commands after a `GAME OVER` are to be ignored). If all the bars are dropped and there was no `GAME OVER`, the credits gained so far is to be given.

Example

Sample input

```
2
6 7 3 6
Ed
EABrd
EABrrd
BBrrrrd
BDBrrrrrd
ABCrrrsd
2 4 3 2
ABCllrrd
ABCrrrrd
```

Correct output for the sample input

```
6
GAME OVER 0
```