![apac test — powered by Google Code Jam]

Problem D. **gSnake**

This contest is open for practice. You can try every problem as many times as you like, though we won't keep track of which problems you solve. Read the Quick-Start Guide to get started.
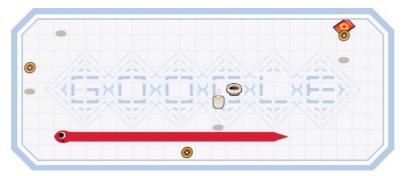
Small input
13 points   [Solve D-small]

Large input
19 points   [Solve D-large]

Problem

Alex is a huge fan of the Snake game.

Note: This Google Doodle does not exactly match the rules of the Snake game we will describe below. It is only intended to give you a general idea of what the game looks like.

Alex just learned how to program and wants to develop his own version of Snake, with the following rules:

- The game board has **R** rows and **C** columns. The top left cell of the board has coordinates (1, 1), and the bottom right cell has coordinates (**R**, **C**).
- At the start of the game, in every cell with coordinates (r, c) such that r + c is odd, there is one piece of food. No other cells have food.
- The snake's body is always an ordered, connected sequence of one or more cells on the board. The first cell of the sequence is called the "head" of the snake. The second cell (if any) shares an edge (not just a corner) with the first cell, and so on. The last cell in the sequence is called the "tail" of the snake.
- The snake's head is always facing in one of four directions: left, up, right, or down.
- At the start of the game, the snake is at cell (1, 1) and has a length of one (that is, the snake consists of only a head), and the head faces right.
- At each integer time (1 second, 2 seconds, etc.), the head of the snake will move into the adjacent cell that its head is facing toward. The board is **cyclic**, i.e., trying to move off an edge will cause the head to appear on the opposite edge of the board. For example, if the snake is at (1, **C**) and its head is facing right, the head will next move to (1, 1). If the snake is at (1, **C**) and its head is facing up, the head will next move to (**R**, **C**).
- When the snake's head moves into a cell with no food, the snake does not grow. The snake's second cell (if any) moves to the place where the snake's head was, the snake's third cell (if any) moves to the place where the second cell was, and so on.
- When the snake's head moves into a cell with a piece of food, it eats the food (meaning that cell no longer has food), and grows its body. A new head is created in the cell where the food was. The cell that was the snake's head becomes the snake's second cell, the cell that was the snake's second cell (if any) becomes the snake's third cell, and so on.
- If, after a move is complete, the snake's head is in the same place as one of another of its cells, the snake dies and the game ends immediately. (Note that if the snake's head moves toward a cell where its tail was, the game will not end, because the tail will move out of the way before the move is complete.)
- In the game, the player can let the snake perform some turn actions. Each action A_i will happen between the T_ith and T_i+1 th seconds. There are two possible actions: "L" and "R". An "L" action will turn the head 90 degrees to the left, so, for example, if the snake had been facing down before, it would face right after. An "R" action will turn the head 90 degrees to the right, so, for example, if the snake had been facing down before, it would face left after.
- The game has a time limit: it will end after the move on the 10^9th second is complete (if the game has even gone on that long!)

To test the game, Alex has written a series of TURN actions. Your task is to simulate that series of actions, and tell Alex the final length of the snake when the game is over. Remember that the game can end either because the snake's

head and another cell of its body are in the same place after a move is complete, or because time runs out. In the former case, you should count both the head and the overlapping cell of its body as two separate cells, for the purpose of determining length.

Input

The first line of the input gives the number of test cases, **T**. **T** test cases follow. Each test cases starts with three integers **S**, **R**, and **C**, where **S** gives the number of turn actions and **R** and **C** represent the number of rows and columns of the board. **S** lines follow; the ith of these lines has an integer X_i, then a character A_i that is either L or R. Each of these lines corresponds to performing an action between X_ith and X_i+1 th seconds. It's guaranteed that the actions are given in time order and there will never be more than one action between the same two seconds. However, you should note that the game may end before the snake gets to execute all of these actions.

Output

For each test case, output one line containing "Case #x: y", where x is the test case number (starting from 1) and y is the length of the snake when the game is over.

Limits

$1 \le$ **T** ≤ 10.

Small dataset

$1 \le$ **R, C** ≤ 100;
$1 \le$ **S** ≤ 100;
$1 \le X_i \le 2000$.

Large dataset

$1 \le$ **R, C** ≤ 100000;
$1 \le$ **S** ≤ 100000;
$1 \le X_i \le 1000000$.

Sample

Input	Output
2	Case #1: 3
3 3 3	Case #2: 5
1 R	
2 L	
3 R	
5 3 3	
2 R	
4 R	
6 R	
7 R	
8 R	