
Problem A. Robot Cleaner I

Input file: **standard input**
Output file: **standard output**
Time limit: 1 second
Memory limit: 256 megabytes

Tired of tidying up his room, BaoBao decides to invent a robot cleaner to help him do the cleaning.

Let's consider BaoBao's room as a grid with n rows and m columns, where the cell in the i -th row and the j -th column is denoted as (i, j) . Each cell belongs to one of the following three types:

- Type 0: This cell is empty;
- Type 1: This cell is a wall;
- Type 2: This cell contains a piece of litter.

As we know, rooms are surrounded by walls. So it's guaranteed that for all $1 \leq i \leq n$, both $(i, 1)$ and (i, m) are walls; And for all $1 \leq j \leq m$, both $(1, j)$ and (n, j) are walls.

After days of hard work, BaoBao successfully creates his very first robot cleaner. The robot is equipped with five sensors, four wheels, a robot arm, and a very simple controller.

The sensors can detect the type of the cell the robot is currently in, as well as the types of the four neighboring cells. That is to say, if the robot is in cell (i, j) , it can know the types of five cells: (i, j) , $(i - 1, j)$, $(i + 1, j)$, $(i, j - 1)$ and $(i, j + 1)$.

The controller accepts a string s consisting of exactly 243 ($243 = 3^5$) characters as the program, where each character represents an instruction, and controls the robot according to the program and the values returned from the sensors. We now list the valid instructions below, and we assume that the robot is currently in cell (i, j) .

- U: If $(i - 1, j)$ is not a wall, move the robot from (i, j) to $(i - 1, j)$. Otherwise do nothing;
- D: If $(i + 1, j)$ is not a wall, move the robot from (i, j) to $(i + 1, j)$. Otherwise do nothing;
- L: If $(i, j - 1)$ is not a wall, move the robot from (i, j) to $(i, j - 1)$. Otherwise do nothing;
- R: If $(i, j + 1)$ is not a wall, move the robot from (i, j) to $(i, j + 1)$. Otherwise do nothing;
- P: If (i, j) contains a piece of litter, the robot will pick up the litter. Otherwise do nothing. Note that after picking up the litter, (i, j) becomes an empty cell;
- I: Do nothing.

The controller works as follows. Note that we still assume that the robot is currently in cell (i, j) , and we denote $t(i, j)$ as the type of cell (i, j) .

1. Calculate $x = 3^4 \times t(i, j) + 3^3 \times t(i - 1, j) + 3^2 \times t(i + 1, j) + 3 \times t(i, j - 1) + t(i, j + 1)$;
2. Read the $(x + 1)$ -th character in s as the instruction and execute it. After that, go back to step 1.

Given the map of BaoBao's room, the program string and the starting position of the robot, please calculate the number of pieces of litter the robot can pick up after executing k instructions.

Input

There are multiple test cases. The first line of the input contains an integer T , indicating the number of test cases. For each test case:

the same line in our input data. We have to split them into different lines here because they're too long to fit into one line of the printed problemset. See the contest website for a more precise input example.