F	Jimmy Move	
	Input	**f.in**
	Output	**Standard Output**

Jimmy is a pretty dump robot. It can only move in 4 directions, north, east, south and west. We will put Jimmy in a square maze and let it move as the direction says. Each block of the maze tells the direction that Jimmy must follow; N, E, S, and W for north, east, south and west, respectively.

After a number of moving, there are three possible results. Jimmy either loop in the maze, stay in the maze or exit from the maze. If Jimmy ever walks into the previous positions, it's a loop. If Jimmy never walks into the previous positions and still stays in the maze, it's a stay. If Jimmy at some point exits the maze, it's exit. All of these results, we also need to know the final position of Jimmy. Jimmy is pretty dump, remember? So even if Jimmy goes into the loop, he still keeps moving.

Please note that the row and column index in the maze starts with zero (0).

Input

The first line is an integer n, 2<= n <=50, represents the size of the maze.

The next n lines are a description of the maze, all are upper case characters.

The next m lines are the test case. Each line contains three integers; r, c and p, separated by a space.

r is a starting row index of Jimmy, 0 <= r < n

c is a starting column index of Jimmy, 0<= c < n

and p is a number of step that Jimmy has to move, 1 <= p <= 100

The input ends with three zeroes (0 0 0).

Output

Each test case, verify Jimmy's result as ‘exit’, ‘stay’ or ‘loop’ and tell the last row and column index of Jimmy in the form ‘(row,column)’. There is no space in the output.

You must display the results in an exact form as shown in the sample output below.

Sample Input	Sample Output
5 NSSSW NSSWE SSSWS SSEWE EEESW 0 1 7 0 0 2 0 4 5 0 2 6 1 0 3 2 0 5 2 0 6 4 4 1 4 4 3 0 0 0	exit(4,3) exit(0,0) stay(3,2) loop(3,3) exit(0,0) stay(4,3) exit(4,3) stay(4,3) exit(4,3)