

## Problem H. Hovercraft

Input file: *standard input*  
Output file: *standard output*  
Time limit: 8 seconds  
Memory limit: 256 mebibytes

You are controlling a hovercraft. Hovercraft is located on a secret base. This base is divided into  $n \times m$  sectors. Rows are numbered with consecutive integers from 1 to  $n$  (from top to bottom), columns — from 1 to  $m$  (from left to right). Each sector is either empty or occupied with a wall. Also, there are  $k$  commutators located at some empty sectors. There is no sector with more than one commutator in it. Each commutator can be either enabled or disabled.

You need to make a list of 12 commands to the hovercraft, and it will execute these commands one by one in the same order. Each of these commands should be one of a following type:

- U — move up: move from sector  $[r, c]$  to sector  $[r - 1, c]$ ;
- D — move down: move from sector  $[r, c]$  to sector  $[r + 1, c]$ ;
- L — move left: move from sector  $[r, c]$  to sector  $[r, c - 1]$ ;
- R — move right: move from sector  $[r, c]$  to sector  $[r, c + 1]$ ;
- S — switch commutator state: enable commutator if it was disabled, and disable otherwise;
- F — call function: execute a predefined list of 8 commands one by one;
- N — no operation: just do nothing.

Hovercraft ignores movement operation if it is impossible to perform it without hitting a wall or violating the secret base borders.

Hovercraft ignores switching operation if there is no commutator in the current cell.

Each function call has the same list of commands, and it is also composed by you. This list should contain exactly 8 commands of the above types. It means that the function can be called recursively.

Your goal is to make all  $k$  commutators enabled at some moment of time. That means your task is to compose a list of 12 commands and a function of 8 commands to make hovercraft enable all  $k$  commutators at the same time. It does not matter what will happen after this moment of time: hovercraft can move indefinitely or even disable some commutators.

### Input

The first line of input contains three integers  $n$ ,  $m$  and  $k$  — the number of rows and columns of the secret base and the number of commutators respectively ( $2 \leq n, m \leq 20$ ,  $1 \leq k \leq 5$ ).

The second line of input contains two integers  $x_s$  and  $y_s$  — row and column number of the initial hovercraft position ( $1 \leq x_s \leq n$ ,  $1 \leq y_s \leq m$ ).

Each of the following  $n$  lines denotes a row of the secret base and contains  $m$  characters. Each character is either '.' (empty sector), 'X' (sector with wall), 'E' (enabled commutator) or 'D' (disabled commutator).

It is guaranteed that the hovercraft starts in a sector without a wall, and also that at least one valid answer exists for the given input.

### Output

The first line of output should contain exactly 8 characters denoting commands of function.

The second line of output should contain exactly 12 characters denoting the main list of commands.

Each of these characters should be either 'U', 'D', 'L', 'R', 'S', 'F' or 'N'.

If there are several possible answers, output any one of them.

## Examples

| standard input   | standard output          |
|--|--------------------------|
| 5 6 3<br>1 3<br>.E....<br>XXXXX.<br>...D..<br>.XXXXX<br>.....D | UUUDDRSF<br>RRRDDLLSLLLF |
| 3 4 2<br>3 3<br>E...<br>.XX.<br>.X.E                           | NNNNNNNN<br>RSUULLLSNMMM |
| 4 6 3<br>4 6<br>DX....<br>.X.XX.<br>.D..X.<br>....XD           | NSLUUUSN<br>NFLNLLDDNLFS |

## Note

There is no need to use function in the second sample. In this case the function can be composed of arbitrary commands.

After performing the second function call of the third sample, all the commutators are enabled. That is why the output is correct despite the fact that the last command disables one of the commutators.