D. Pawn

On some square in the lowest row of a chessboard a stands a pawn. It has only two variants of moving: upwards and leftwards or upwards and rightwards. The pawn can choose from which square of the lowest row it can start its journey. On each square lay from 0 to 9 peas. The pawn wants to reach the uppermost row having collected as many peas as possible. As there it will have to divide the peas between itself and its k brothers, the number of peas must be divisible by $k + 1$. Find the maximal number of peas it will be able to collect and which moves it should make to do it.

The pawn cannot throw peas away or leave the board. When a pawn appears in some square of the board (including the first and last square of the way), it necessarily takes all the peas.

Input

The first line contains three integers n, m, k $(2 \le n, m \le 100, 0 \le k \le 10)$ — the number of rows and columns on the chessboard, the number of the pawn's brothers. Then follow n lines containing each m numbers from 0 to 9 without spaces — the chessboard's description. Each square is described by one number — the number of peas in it. The first line corresponds to the uppermost row and the last line — to the lowest row.

Output

If it is impossible to reach the highest row having collected the number of peas divisible by $k + 1$, print -1.

Otherwise, the first line must contain a single number — the maximal number of peas the pawn can collect given that the number must be divisible by $k + 1$. The second line must contain a single number — the number of the square's column in the lowest row, from which the pawn must start its journey. The columns are numbered from the left to the right with integral numbers starting from 1. The third line must contain a line consisting of n - 1 symbols — the description of the pawn's moves. If the pawn must move upwards and leftwards, print L, if it must move upwards and rightwards, print R. If there are several solutions to that problem, print any of them.

Examples

input

```
3 3 1
123
456
789
```

output

```
16
2
RL
```

input

```
3 3 0
123
456
789
```

output

```
17
3
LR
```

input

```
2 2 10
98
75
```

output

```
-1
```