

Given a board of R rows \times C columns. The rows are numbered from 0 to $R-1$, the columns are numbered from 0 to $C-1$.

There are N tiles on the board (each tile occupies exactly 1 cell), numbered from 1 to N .

Given Q queries, each query has the form " i direction", where $1 \leq i \leq N$ and $direction \in \{"up", "down", "left", "right"\}$ means slide the tile numbered i 1 unit in that direction. If there are another tile at this cell, move that tile 1 unit in that direction... (do not move anything if the operation would push one tile out of the board's edge)

Input

There are multiple (≤ 2) tests, process tests until EOF is seen.

- Line 1: 5 numbers R, C, N, Q, B ($1 \leq B \leq 10^8$)
- N subsequent lines: each line has the form " $r[i] c[i]$ " specifies the location of the tile numbered i . ($1 \leq i \leq N$, it's guaranteed that there are no two different tiles in the same location)
- Q subsequent lines: each line consistent of a query, with the format specified above.

Output

Number the cells in the board top-to-bottom, left-to-right.

Let a_i ($0 \leq i \leq R \times C - 1$) be the number of the tile in the cell i , or 0 if there's no tile.

Print

$$(a_0 B^{R \times C - 1} + a_1 B^{R \times C - 2} + \dots + a_{R \times C - 1} B^0) \bmod 998244353$$

Subtask

- Subtask 1 (10%): There are only "up"/"down" query types.
- Subtask 2 (30%): There are only "left"/"right" query types.
- Subtask 3 (60%): No restriction.

Sample Input	Sample Output
3 3 3 3 10 0 0 0 1 1 0 1 right 1 right 3 up	312000000

Note (operation 2 is not doing anything, otherwise, tile 2 is out of the board)

Initial state of the board:

1	2	
3		

Final state of the board:

3	1	2