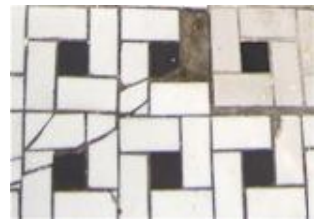


Today is Tilliby's 9-th birthday. He has been receiving all kinds of gifts from people like glow-in-the-dark stickers, electronic calculator, a new toothbrush and so forth. However, this funny looking puzzle involving domino tiles caught his attention. The rules were as follows — the puzzle consisted of a rectangular grid of  $R$ -by- $C$  cells, and it must be completely filled by  $2 \times 1$  sized tiles, of which there is sufficient supply. No two tiles may overlap. To complicate things even more, certain cells of the grid is marked unusable, which must not be covered by any of the tiles. All other cells, however, must be covered by exactly one tile. Now in spite of all these complications, this was an easy exercise for Tilliby. However, when it came to counting the number of ways this can be solved, even his new and shiny calculator could not help him. Can you?



## Input

The input consists of at most 100 test cases. Each test case starts with a line containing 3 integers,  $R$ ,  $C$  and  $N$ . This line will be followed by  $N$  other lines, each containing two integers,  $r_i$  and  $c_i$ , where  $r_i$  is the row position of the  $i$ -th unusable cell and  $c_i$  is the column position. All these input integers will be within the following ranges:  $1 \leq R \leq 4$ ,  $1 \leq C \leq 100000000$ ,  $0 \leq N \leq 100$ ,  $0 \leq r_i < R$ ,  $0 \leq c_i < C$ .

The last test case will be followed by a single line containing three '0' (zeroes) which should not be processed.

## Output

The output for each test case should consist of one single line of the form 'Case  $c$ :  $x$ ', where  $c$  is the serial number of the test case starting from 1, and  $x$  is the number of ways the specified tiling can be performed. Since this number can be very large, output its value in mod10000007.

## Sample Input

```
2 2 0
2 4 0
2 4 2
0 0
0 3
0 0 0
```

## Sample Output

```
Case 1: 2
Case 2: 5
Case 3: 1
```