

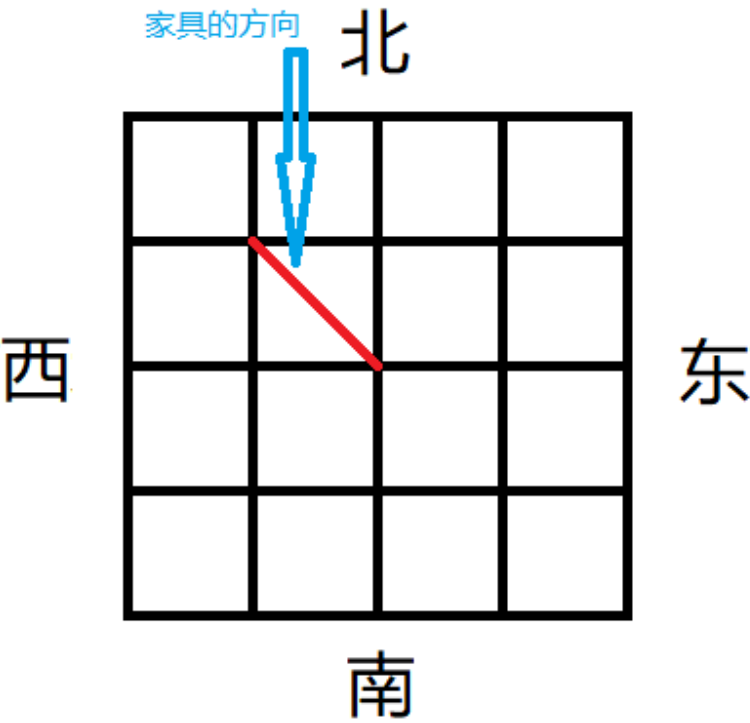
# 走廊

时间限制：1.5s 内存限制：1.0GB  
输入文件名：corridor.in 输出文件名：corridor.out

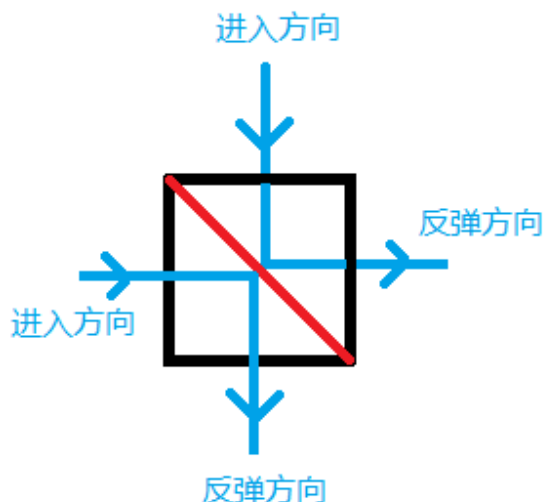
## 题目描述

GreenDuck 买到了最新的扫地机器人——RobotDuck。为了更高效地清扫房间，GreenDuck 决定分析一下房间的布局 and RobotDuck 的清扫机制。

GreenDuck 的房间可以看成是  $n$  行  $m$  列的网格，每个格子中要么是空的，要么放了一件家具。同时他惊奇地发现，每个家具都很薄(可以看成是一条线段)，两个端点分别占据了一个格子的西北角和东南角！



RobotDuck 在清扫时会采取这样一种机制：刚开始，可以给它设定一个方向(向南或向东)，接着它将一直沿着直线行进。如果碰到一个家具，那么会反弹。具体方式如下图所示。



GreenDuck 一开始得知, 在自己的房间里有  $k$  件家具, 第  $i$  件家具在第  $x_i$  行,  $y_i$  列。接下来, 他会进行  $Q$  次测试。

第一种测试, 是在一个没有家具的格子上一件家具(方向保持一致)。这种测试后, 他不会拿走任何家具。

第二种测试, 是将 RobotDuck 贴着墙壁然后释放。具体来说, 如果 RobotDuck 放在北面(对应了图中矩形的上边界), 那么它会从第一行某个格子的上边界的中心出发, 面向南面行进。如果 RobotDuck 放在西面(对应了图中矩形的左边界), 那么它会从第一列某个格子的左边界的中心出发, 面向东面行进。在行进过程中, 它遵守清扫的机制。

GreenDuck 想知道, 在每次第二种测试时, RobotDuck 在反弹恰好  $q$  次后会在哪个格子里。请你告诉他。

---

## 输入格式

第一行四个数字,  $type, n, m, k$ , 分别表示数据类型(你可能不需要), 行数, 列数, 一开始有的家具件数。

接下来  $k$  行, 每行两个整数  $x_i, y_i$ , 分别表示这件家具的行数和列数。

接下来一行一个整数  $Q$ , 表示测试的次数。

接下来  $Q$  行, 首先输入一个数字  $w$ 。

若  $w$  为 1, 则会有两个数字  $x, y$ , 分别表示新添的家具的行数和列数。

若  $w$  为 2, 则会有三个数字  $x, y, q$ 。若  $x = 0$ , 表示 RobotDuck 从第一行第  $y$  列格子的上边界的中心面向南行进。否则  $y = 0$ , 表示 RobotDuck 从第一列第  $x$  行格子的左边界的中心面向东行进。 $q$  表示反弹的次数。

## 输出格式

对于每个  $w = 2$  的操作，输出一行两个数字  $x$  和  $y$  分别表示所在格子的行数和列数。

特别地，若 RobotDuck 在第  $q$  次反弹之前就碰到了墙壁，若墙壁是第  $n$  行格子的下边界，输出" $n + 1$  列数"，否则输出"行数  $m + 1$ "。

## 样例输入 1

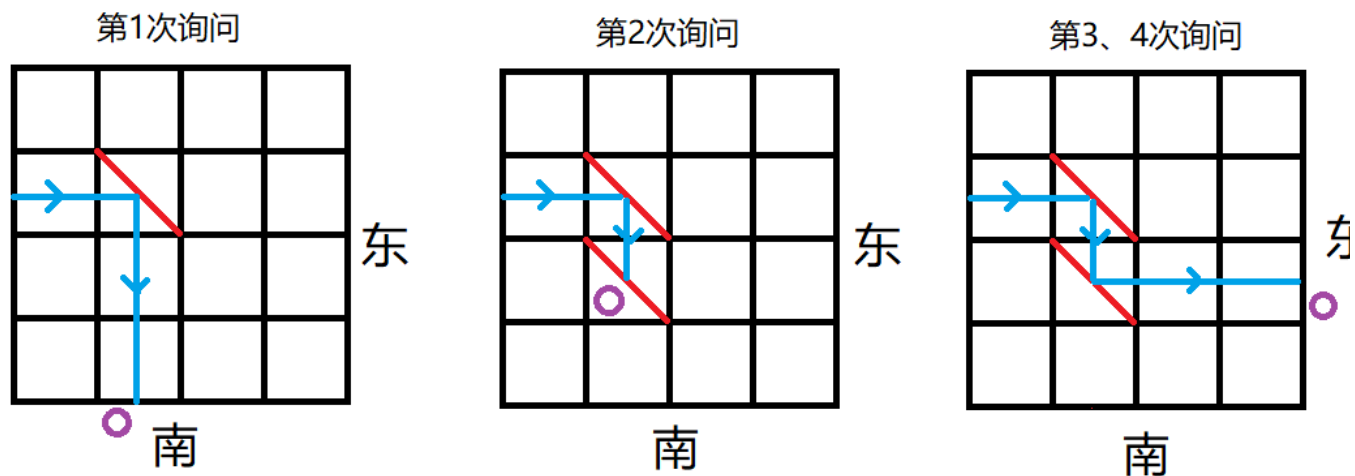
```
0 4 4 1
2 2
5
2 2 0 3
1 3 2
2 2 0 2
2 2 0 3
2 2 0 4
```

## 样例输出 1

```
5 2
3 2
3 5
3 5
```

## 样例 1 解释

下图中紫色的圆圈表示最终所到达的格子。



## 样例 2

见下发文件下的 `corridor2.in` 和 `corridor2.ans` 。

## 数据范围

对于所有数据，家具的坐标不会重复，每次第二种操作要么  $x = 0$  要么  $y = 0$ 。

$1 \leq q \leq 200000$ 。

测试点编号	$type$	$n, m$	$k + Q$	特殊性质
1 ~ 5	$= 0$	$\leq 100$	$\leq 2000$	无
6 ~ 11	$= 1$	$\leq 20000$	$\leq 150000$	询问中不会出现1号操作
12 ~ 17	$= 2$	$\leq 50000$	$\leq 150000$	反弹次数之和不超过 $10^7$ (不是 $q$ 之和)
18 ~ 25	$= 3$	$\leq 50000$	$\leq 150000$	无