



## 全国信息学奥林匹克联赛（NOIP2018）复赛模拟

Cssyz zzs

## 一、题目概况

中文题目名称	矩阵游戏	跳房子	优美序列
英文题目名称	game	jump	sequence
可执行文件名	game	jump	sequence
输入文件名	game.in	jump.in	sequence.in
输出文件名	game.out	jump.out	sequence.out
每个测试点时限	1 秒	3 秒	1 秒
测试点数目	20	20	25
每个测试点分值	5	5	4
附加样例文件	有	有	有
题目类型	传统	传统	传统

## 二、提交源程序文件名

对于 pascal 语言	game.pas	jump.pas	sequence.pas
对于 C 语言	game.c	jump.c	sequence.c
对于 C++语言	game.cpp	jump.cpp	sequence.cpp

## 三、编译命令（不包含任何优化开关）

对于 pascal 语言	fpc game.pas	fpc jump.pas	fpc sequence.pas
对于 C 语言	gcc -o game game.c -lm -O2	gcc -o jump jump.c -lm -O2	gcc -o sequence sequence.c -lm -O2
对于 C++语言	g++ -o game game.cpp -lm -O2	g++ -o jump jump.cpp -lm -O2	g++ -o sequence sequence.cpp -lm -O2

## 四、运行内存限制

内存上限	512M	512M	512M
------	------	------	------

## 五、注意事项

- 1、文件名（程序名和输入输出文件名）必须使用小写。
- 2、C/C++中函数 main()的返回值类型必须是 int，程序正常结束时的返回值必须是 0。



# 1. 矩阵游戏

(game.pas/c/cpp)

## 【问题描述】

LZK 发明一个矩阵游戏，大家一起来玩玩吧，有一个  $N$  行  $M$  列的矩阵。第一行的数字是  $1, 2, \dots, M$ ，第二行的数字是  $M+1, M+2, \dots, 2*M$ ，以此类推，第  $N$  行的数字是  $(N-1)*M+1, (N-1)*M+2, \dots, N*M$ 。

例如， $N=3, M=4$  的矩阵是这样的：

1	2	3	4
5	6	7	8
9	10	11	12

对于身为智慧之神的 LZK 来说，这个矩阵过于无趣。于是他决定改造这个矩阵，改造会进行  $K$  次，每次改造会将矩阵的某一行或某一列乘上一个数字，你的任务是计算最终这个矩阵内所有数字的和，输出答案对  $10^9+7$  取模。

## 【输入】

第一行包含三个正整数  $N, M, K$ ，表示矩阵的大小与改造次数。接下来的行，每行会是如下两种形式之一：

$R \ X \ Y$ ，表示将矩阵的第  $X$  ( $1 \leq X \leq N$ ) 行变为原来的  $Y$  ( $0 \leq Y \leq 10^9$ ) 倍。

$S \ X \ Y$ ，表示将矩阵的第  $X$  ( $1 \leq X \leq M$ ) 列变为原来的  $Y$  ( $0 \leq Y \leq 10^9$ ) 倍。

## 【输出】

输出一行一个整数，表示最终矩阵内所有元素的和对  $10^9+7$  取模的结果。

## 【输入输出样例】

样例 1		样例 2	
game.in	game.out	game.in	game.out
3 4 4	94	2 4 4	80
R 2 4		S 2 0	
S 4 1		S 2 3	
R 3 2		R 1 5	
R 2 0		S 1 3	

样例一的解释：操作结束之后矩阵会变成这样：

1	2	3	4
0	0	0	0
18	20	22	24

## 【数据范围】

40%的数据满足：  $1 \leq N, M \leq 1000$ ；

80%的数据满足：  $1 \leq N, M \leq 1000000, 1 \leq K \leq 1000$ ；

100%的数据满足：  $1 \leq N, M \leq 1000000, 1 \leq K \leq 100000$ 。



## 2. 跳房子

(jump.pas/c/cpp)

### 【问题描述】

跳房子，是一种世界性的儿童游戏，也是中国民间传统的体育游戏之一。

跳房子是在  $N$  个格子上的，CYJ 对游戏进行了改进，该成了跳棋盘，改进后的游戏是在一个  $N$  行  $M$  列的棋盘上进行，并规定从第一行往上可以走到最后一行，第一列往左可以走到最后一列，反之亦然。每个格子上有一个数字。

在这个棋盘左上角  $(1, 1)$  放置着一枚棋子。每次棋子会走到右、右上和右下三个方向格子中对应上数字最大一个。即任意时刻棋子都只有一种走法，不存在多个格子同时满足条件。

现在有两种操作：

move  $k$  将棋子前进  $k$  步。

change  $a\ b\ e$  将第  $a$  行第  $b$  列格子上的数字修改为  $e$ 。

请对于每一个 move 操作输出棋子移动完毕后所处的位置。

### 【输入】

第一行包含两个正整数  $N, M$  ( $3 \leq N, M \leq 2000$ )，表示棋盘的大小。

接下来  $N$  行，每行  $M$  个整数，依次表示每个格子中的数字  $a[i, j]$  ( $1 \leq a[i, j] \leq 10^9$ )。

接下来一行包含一个正整数  $Q$  ( $1 \leq Q \leq 5000$ )，表示操作次数。

接下来  $m$  行，每行一个操作，其中  $1 \leq a \leq N, 1 \leq b \leq M, 1 \leq k, e \leq 10^9$ 。

### 【输出】

对于每个 move 操作，输出一行两个正整数  $x, y$ ，即棋子所处的行号和列号。

### 【输入输出样例】

jump.in	jump.out
4 4	4 2
1 2 9 3	1 3
3 5 4 8	1 4
4 3 2 7	
5 8 1 6	
4	
move 1	
move 1	
change 1 4 100	
move 1	

### 【数据范围】

10%的数据满足： $3 \leq N, M \leq 50, Q \leq 5000, k \leq 10$ ；

20%的数据满足： $3 \leq N, M \leq 200, Q \leq 5000, k \leq 5000$ ；

另有20%的数据满足： $3 \leq N, M \leq 200, Q \leq 5000, k \leq 10^9$ ；

100%的数据满足： $3 \leq N, M \leq 2000, Q \leq 5000, e, k \leq 10^9$ ；



### 3. 优美序列

(sequence.pas/c/cpp)

#### 【问题描述】

Lxy 养了  $N$  头奶牛，他把  $N$  头奶牛用  $1..N$  编号，第  $i$  头奶牛编号为  $i$ 。为了让奶牛多产奶，每天早上他都会让奶牛们排成一排做早操。奶牛们是随机排列的。在奶牛排列中，如果一段区间  $[L, R]$  中的数从小到大排列后是连续的，他认为这段区间是优美的。比如奶牛排列为：(3, 1, 7, 5, 6, 4, 2)，区间  $[3, 6]$  是优美的，它包含 4, 5, 6, 7 连续的四个数，而区间  $[1, 3]$  是不优美的。Lxy 的问题是：对于给定的一个区间  $[L, R]$  ( $1 \leq L \leq R \leq N$ )，他想知道，包含区间  $[L, R]$  的最短优美区间，比如区间  $[1, 3]$  的最短优美区间是  $[1, 7]$ 。

#### 【输入】

第一行为一个整数  $N$ ，表示奶牛的个数。  
第二行为 1 到  $N$  的一个排列，表示奶牛的队伍。  
第三行为一个整数  $M$ ，表示有  $M$  个询问。  
后面有  $M$  行，每行有两个整数  $L, R$  表示询问区间。

#### 【输出】

输出为  $M$  行，每行两个整数，表示包含询问区间的最短优美区间。

#### 【输入输出样例】

样例 1		样例 2	
sequence.in	sequence.out	sequence.in	sequence.out
7	3 6	10	1 4
3 1 7 5 6 4 2	7 7	2 1 4 3 5 6 7 10 8 9	3 7
3	1 7	5	3 7
3 6		2 3	3 10
7 7		3 7	7 10
1 3		4 7	
		4 8	
		7 8	

#### 【数据范围】

15%的数据满足：  $1 \leq N, M \leq 15$ ；  
50%的数据满足：  $1 \leq N, M \leq 1000$ 。



100%的数据满足：  $1 \leq N, M \leq 100000$ 。