

# NOIP 模拟题

Cxt

August 5, 2018

题目名称	听我说, 海蜗牛	触手不及	有所失
提交程序名	connect.cpp	expression.cpp	travel.cpp
空间限制	512MB	512MB	512MB
时间限制	1s	1s	1s
测试点数	10	20	20
测试点分值	10	5	5
题目类型	传统型	传统型	传统型
编译命令	-O2 -lm	-O2 -lm	-O2 -lm

# 1 听我说, 海蜗牛

(connect.cpp/c/pas)

## 1.1 Background

风平浪静的大海开始泛起波澜。

## 1.2 Description

在广阔的大海沿岸有  $n$  个港口, 其两两之间都可以通航. 但是最近, 风平浪静的大海开始泛起波澜, 使得其中  $m$  对港口之间的航路中断了.

这个月有  $q$  天, 每天都有一些港口是开放的. 定义两个开放的港口联通, 为它们可以只经过开放的港口以及未间断的航路相到达.

在这种情况下, 小 M.M.T. 希望能够知道对于每一天, 那些开放的港口被划分为了多少个联通块. 她会告诉你每天开放的港口编号, 由你来求出答案.

## 1.3 Input

第一行三个数  $n, m, q$ , 意义如题面.

之后  $m$  行, 每行两个数  $x, y$  表示  $x$  和  $y$  的航路中断了.

保证无重边无自环.

之后  $2 * q$  行, 对于第  $2 * i$  行输入一个数  $k$ , 对于第  $2 * i + 1$  行输入  $k$  个数, 表示这天开放的港口数.

## 1.4 Output

$q$  行, 每行一个数表示这天联通块的个数.

## 1.5 Sample

input1

```
4 4 3
1 4
1 2
1 3
2 3
4
1 2 3 4
```

3  
1 2 3  
2  
1 2

output1

2  
3  
2

1.6 Limit

CaseNum	$n \leq$	$m \leq$	$q \leq$	$\Sigma k$
1	10	20	10	100
2	100	200	100	10000
3	5000	10000	5000	100000
4				
5	50000	100000	50000	100000
6				
7				
8	100000	200000	100000	200000
9				
10				

## 2 触手不及

(expression.cpp/c/pas)

### 2.1 Background

—我一直注视着你, 似近, 似远。但你永远, 看不见我.....

### 2.2 Description

为了避免流露出自己的感情伤害别人, 小 M.M.T. 决定通过一个表达式来传递心意.

给出一个等式.

等式左边是一个 *int* 范围内的数, 等式右边是一个合法的 c++ 表达式.

例如:  $233 = 66 * 4 - 31$

保证等式右边只包含数字  $x$  ( $x \in [0, p)$ ,  $p$  是给定的质数), 加号, 减号, 乘号, 除号, 左右括号.

保证等式中没有任何空格, tab 等不可见字符. 而且保证合法 (即拖到 c++ 中可以通过编译)

但是遗憾的是, 因为一些原因, 该等式不保证成立.

于是, 小 M.M.T. 希望知道, 在模  $p$  意义下, 她的表达式的每个数字  $x$ , 需要变成多少才能使等式成立.

保证原不等式不存在除 0, 你需要保证把数字变成  $x$  之后等式仍然不会除 0.

如果无论  $x$  取多少都不能使等式成立, 则输出 "No Solution".

如果无论  $x$  取多少都能使等式成立, 则输出 "-1".

### 2.3 Input

输入共两行,

第一行两个整数  $n, p$  ( $n, p \geq 1$ ) 表示等式右边共会出现  $n$  个整数.

第二行给出长为  $len$  的等式.

### 2.4 Output

$n$  行, 每行一个数  $ans$  ( $ans \in [-1, p)$ ) 或者 "No Solution".

### 2.5 Sample

input1

3 11  
 $6=1*0+4$

**output1**

No Solution  
2  
6

**input2**

5 11  
 $10=2*(3+5-1)/1$

**output2**

3  
1  
3  
3  
8

## 2.6 Limit

CaseNum	$len \leq$	$p \leq$	特殊性质
1	10	10	无
2			
3	100000	10000	等式保证成立
4	100000	10000	等式中符号只包含'+','-'
5			
6	100000	10000	等式中符号只包含'*','/'
7			
8	100000	10000	等式中符号不包含'(',')'
9			
10			
11			
12	5000	2000	无
13			
14			
15			
16	5000000	500000	无
17			
18			
19			
20			

## 3 有所失

(travel.cpp/c/pas)

### 3.1 Background

我，希望明天，依旧风平浪静。

### 3.2 Description

时光飞逝，五年过去了，小 M.M.T. 长大成为了一名妙龄少女。

她决定独自探索大海。

大海可以视为一个无限大的二维平面，初始时，小 M.M.T. 在  $(-\infty, -\infty)$ 。

由于技术原因，小 M.M.T. 乘坐的船只拥有两个第一象限点作为参数： $P_{x,y}(p.x > p.y)$ ， $Q_{x,y}(q.x < q.y)$ 。这两个点（向量）构成了一个夹角（包括角的两边），表示每次船只只能向这个夹角内的方向前进。

大海内初始时有  $n$  个岛屿，每个岛屿都在第一象限。小 M.M.T. 的探索计划有  $m$  天，每天小 M.M.T. 会希望从初始点出发到某一个岛屿上探索，或者会发现一个新岛屿。

为了方便，岛屿两两之间（包括新发现的）没有  $x$  相同且  $y$  相同的。

另外，设新发现的岛屿为  $O_{x,y}$ ，由于是新发现的，可以保证：只有不超过  $k$  个点在  $O$  和  $O + P$  所形成直线的左侧<sup>1</sup>，或者只有不超过  $k$  个点在  $O$  和  $O + Q$  所形成直线的右侧。

为了欣赏更多沿路的风景，她希望在行进的过程中经过更多的岛屿。

因为输出方案太麻烦，你可以直接告诉她，她最多能经过的岛屿数目。

### 3.3 Input

第一行两个数  $n, k$ 。

第二行四个数，即两个坐标  $P.x, P.y, Q.x, Q.y$  表示船的两个参数。

之后  $n$  行每行两个数  $P.x, P.y$ ，表示初始就存在的岛屿的位置。

然后首先一行一个数  $m$ 。

接下来  $m$  行每行先输入一个  $opt$

若  $opt = 1$ ，紧跟一个数  $id$  这次要探索的岛屿。保证  $id$  存在。原先存在的岛屿的  $id$  为  $1, 2, 3, \dots$  新发现的岛屿的  $id$  为  $n + 1, n + 2, n + 3, \dots$

若  $opt = 2$  or  $3$ ，紧跟两个数  $O.x, O.y$  表示要插入的新岛屿。

2 表示只有不超过  $k$  个点在  $O$  和  $O + P$  所形成直线的左侧，

3 表示只有不超过  $k$  个点在  $O$  和  $O + Q$  所形成直线的右侧。

---

<sup>1</sup>我们定义一个点  $U$  在左侧为： $P * (U - O) \geq 0$ ，其中  $*$  为叉积。右侧则  $\leq$ 。

### 3.4 Output

若干行, 对于每一个 2 操作, 输出最多能经过多少个点.

### 3.5 Sample

input1

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

output1

```
2
3
4
```

input2

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



output2

1  
2  
3

### 3.6 Limit

CaseNum	$n \leq$	$m \leq$	$k \leq$	特殊性质 1	特殊性质 2
1	10	10	5	所有 $opt = 1$	$P = (1, 0)$ $Q = (0, 1)$
2	10	10			
3	100	100			
4	100	100			
5	1000	1000			
6	1000	1000	20	无	
7	50000	50000			
8	50000	50000			
9	50000	50000			
10	50000	50000			
11	100	100	5	所有 $opt = 1$	无
12	100	100			
13	1000	1000			
14	1000	1000			
15	100000	100000			
16	100000	100000	20	无	
17	100000	100000			
18	100000	100000			
19	100000	100000			
20	100000	100000			

对于所有的数据, 保证输入的数字都是小于  $10^9$  的整数