

# NOIP2015 模拟赛

By nodgd

竞赛时间：不知道

题目名称	无聊的计算	奇怪的队列	仔细的检查
英文名	A	B	C
输入文件名	A.in	B.in	C.in
输出文件名	A.out	B.out	C.out
每个测试点时限	1s	1s	1s
空间限制	256MB	256MB	256MB
测试点数目	10	10	20
每个测试点分值	10	10	5
是否有部分分	无	无	无
题目类型	传统型	传统型	传统型

评测环境：

不知道

温馨提醒：

不知道

## 无聊的计算

### 【问题描述】

nodgd 经常遇到很无聊的计算题，你看，这次又遇到了……

这个题一开始给了 nodgd 两个序列 $a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_m$ 。定义一个二元函数 $f(u, p) = v$ ，其中 $0 \leq v < p$ ，且存在整数 $k$ 使得 $u = v + kp$ 。现在，这道题要计算满足条件 $f(a_i^{b_j}, p) \leq q$ 的有序数对 $(i, j)$ 有多少个。

### 【输入格式】

输入文件 A.in。

第一行两个整数 $p, q$ 。

第二行六个数 $n, a_1, a_2, A, B, C$ ，当 $i \geq 3$ 时， $a_i = A \times a_{i-1} - B \times a_{i-2} - C$ 。

第三行六个数 $m, b_1, b_2, D, E, F$ ，当 $i \geq 3$ 时， $b_i = D \times b_{i-1} + E \times b_{i-2} + F$ 。

### 【输出格式】

输出文件 A.out。

输出一个整数，表示满足条件的 $(i, j)$ 数量。

### 【样例输入】

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

### 【样例输出】

```
6
```

### 【样例解释】

根据输入可以算出 $a_3 = -2, b_3 = 15$ 。

满足条件的 $(i, j)$ 有 $(1,1), (2,1), (3,1), (1,3), (2,3), (3,3)$ 。

### 【数据范围】

对于 20%的数据， $n, m \leq 6$ ；

对于 40%的数据， $n, m \leq 1000$ ；

对于 60%的数据， $n, m \leq 5000$ ；

对于 100%的数据， $2 \leq n, m \leq 10^6, 5 \leq p \leq 5000, 0 \leq q \leq p - 1$ ，输入的所有其他数都非负且不超过 1000，保证 $p$ 是个质数。

## 奇怪的队列

### 【问题描述】

nodgd 的粉丝太多了，每天都有很多人排队要签名。

今天有 $n$ 个人排队，每个人的身高都是一个整数，且互不相同。很不巧，nodgd 今天去忙别的事情去了，就只好让这些粉丝们明天再来。同时 nodgd 提出了一个要求，每个人都要记住自己前面与多少个比自己高的人，以便于明天恢复到今天的顺序。

但是，粉丝们或多或少都是有些失望的，失望使她们晕头转向、神魂颠倒，已经分不清楚哪一边是“前面”了，于是她们可能是记住了前面比自己高的人的个数，也可能是记住了后面比自己高的人的个数，而且他们不知道自己记住的是哪一个方向。

nodgd 觉得，即使这样明天也能恢复出一个排队顺序，使得任意一个人的两个方向中至少有一个方向上的比他高的人数和他记住的数字相同。可惜 $n$ 比较大，显然需要写个程序来解决，nodgd 很忙，写程序这种事情就交给你了。

### 【输入格式】

输入文件 B.in。

第一行输入一个整数 $n$ ，表示指令的条数。

接下来 $n$ 行，每行两个整数 $a_i, b_i$ ，表示一个人的身高和她记住的数字，保证身高互不相同。

### 【输出格式】

输出文件 B.out。

输出一行，这个队列里从前到后的每个人的身高。如果有多个答案满足题意，输出字典序最小。如果不存在满足题意的排列，输出“impossible”（不含引号）。

### 【样例输入 1】

```
4
4 1
3 1
6 0
2 0
```

### 【样例输出 1】

```
2 4 3 6
```

### 【样例解释 1】

在所给出的答案队列中，第一个人身高为 2，前面有 0 个人比他高，所以他是输入的第 4 个人；第二个人身高为 4，右边有 1 个人比他高，所以他是输入的第 1 个人；第三个人身高为 3，右边有 1 个人比他高，所以他是输入的第 2 个人；第四个人身高为 6，左边有 0 个人比他高，所以他是输入的第 3 个人。

显然，如果排列为“6 3 4 2”也是满足题意的，但是字典序不是最小的。

**【样例输入 2】**

```
6
1 5
8 0
3 1
4 0
2 0
6 0
```

**【样例输出 2】**

```
1 2 4 3 6 8
```

**【数据范围】**

对于 40%的数据， $n \leq 10$ ；  
对于 60%的数据， $n \leq 1000$ ；  
对于 100%的数据， $1 \leq n \leq 100000$ 。

## 仔细的检查

### 【问题描述】

nodgd 家里种了一棵树，有一天 nodgd 比较无聊，就把这棵树画在了一张纸上。另一天 nodgd 更无聊，就又画了一张。

这时 nodgd 发现，两次画的顺序是不一样的，这就导致了原本的某一个节点  $u_0$  在第一幅图中编号为  $u_1$ ，在第二副图中编号为  $u_2$ 。

于是，nodgd 决定检查一下他画出的两棵树到底是不是一样的。nodgd 已经给每棵树的节点都从1到 $n$ 进行了编号，即每棵树有 $n$ 个节点。

如果存在一个1到 $n$ 的排列  $p_1 p_2 \dots p_n$ ，对于第一幅图中的任意一条边  $(u, v)$ ，在第二幅图中都能找到一条边  $(p_u, p_v)$ ，则认为这两幅图中的树是一样的。

### 【输入格式】

输入文件 C.in。

第一行一个整数 $n$ ，表示节点的总数。

接下来 $n - 1$ 行，每行两个整数，表示第一幅图中的每一条边。

接下来 $n - 1$ 行，每行两个整数，表示第二幅图中的每一条边。

### 【输出格式】

输出文件 C.out。

如果两幅图的树是一样的，第一行输出“YES”，接下来1行输出一个1到 $n$ 的排列  $p_1 p_2 \dots p_n$ ，两个数之间用空格间隔。当多个排列都满足题意时，你可以随便输出一个。

如果两幅图的树是不一样的，只输出一行“NO”。

注意输出的时候不要加引号。

### 【样例输入 1】

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

### 【样例输出 1】

```
YES
1 3 2
```

### 【样例解释 1】

肉眼可见，1-2-3 和 1-3-2 显然是一样的两棵树。不过这可能不是唯一的符合题意的排列。

【样例输入 2】

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

【样例输出 2】

NO

【样例解释 2】

显然无论如何两棵树都是不一样的。

【数据范围】

若无特殊说明，则答案是 YES。

测试点编号	$n =$	特殊的附加条件
1	8	无
2	10	
3	16	
4	20	
5	50	
6	100	
7	300	
8	1000	
9	2000	
10	5000	
11	10000	
12		
13	100000	两棵树各自度数最大的点的度数 = 3， 且恰好有 2 个
14		两棵树各自度数最大的点的度数 = 4， 且恰好有 2 个
15		都是以 1 号点为根的二叉树， 且存在一种方案使得 $p_1 = 1$
16		答案是 NO
17		两棵树如果从 1 号点开始以遍历， 则树高都不超过 20
18		按照“一开始全是独立的节点，每次随机 加一条不构成环的边”的方式随机生成
19		无
20		