

NOIP 提高组模拟赛 1

题目名称	快乐传递政治正确版	嫌疑人	异或
可执行文件名	happy2	suspect	xor
输入文件	happy2.in	suspect.in	xor.in
输出文件	happy2.out	suspect.out	xor.out
每个测试点时限	1s	2s	3s
内存限制	64M	128M	256M
测试点数目	10	20	20
每个测试点分值	10	5	5
结果比较方式	忽略多余的空格和文件尾的空行		
题目类型	传统	传统	传统

注意事项:

1. 比赛时间 3.5 小时
2. 评测在 Windows 下进行, 64 位整数请使用 I64d 输出
3. 评测时忽略多余的空格和制表符
4. 评测时不开任何优化开关
5. 数据范围表格中的数字为对应变量的最大值, 不是准确值
6. 认真读题, 题面可能有疏漏之处, 如有疑问应及时向监考老师询问
7. 部分题目来源或改编自 CF, 赛后给出链接

快乐传递政治正确版 happy2

问题描述

David 有很多好朋友。有些期末季刚结束，有些人很快乐，但有些不太快乐，David 想把快乐传递给每个人，作为心理学大师，他准备了如下计划：

David 的朋友中有 n 个男生和 m 个女生，还有 k 个跨性别者，方便起见，将他们分别编号为 $0, \dots, n-1$ 和 $0, \dots, m-1, 0, \dots, k-1$ ，在第 i 天，David 会邀请编号为 $(i \bmod n)$ 的男生和编号为 $(i \bmod m)$ 的女生还有 $(i \bmod k)$ 的跨性别者共进晚餐（因为 David 同时是程序员，所以从这个计划从第 0 天开始）。共进晚餐的三个人只要至少有一个是快乐的人，另外的人也会变得快乐起来。否则大家的状态不会改变（一旦一个人是快乐的，他就会永远快乐下去）。

现在问题来了，David 想知道他是否能通过这个计划使得所有人都快乐起来呢？

输入格式

第一行一个整数 $T \leq 20$ ，表示该测试点数据组数

每组数据的第一行包含三个整数 n, m, k ，保证至少一类人不为空。

接下来一行第一个整数为 $0 \leq b \leq n$ ，表示目前快乐的男生的数量，接下来 b 个空格隔开的整数 $0 \leq x_i < n$ ，表示快乐的男生的编号。

接下来一行第一个整数为 $0 \leq g \leq m$ ，表示目前快乐的女生的数量，接下来 g 个空格隔开的整数 $0 \leq y_i < m$ ，表示快乐的女生的编号。

接下来一行第一个整数为 $0 \leq t \leq k$ ，表示目前快乐的跨性别者的数量，接下来 t 个空格隔开的整数 $0 \leq z_i < k$ ，表示快乐的跨性别者的编号。

相邻两组数据之间有一行空行

输出格式

假如最终所有人都能变的快乐起来，就输出”Yes”（不带引号），否则输出”No”。

样例输入 1

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

```
2 4 0
1 0
1 2
0
```

```
2 4 1
1 0
1 2
1
```

```
2 3 0
1 0
1 1
0
```

样例输出 1

```
Yes
No
```

Yes

Yes

样例解释

mod 是取模运算的意思.

对于第一组数据:

第 0 天, 0 号男生和 0 号女生吃饭, 因为 0 号女生是快乐的, 所以 0 号男生也变得快乐了;

第 1 天, 1 号男生和 1 号女生吃饭, 因为他们本来就都不快乐, 情况没有变化;

第 2 天, 0 号男生和 2 号女生吃饭, 因为 0 号男生是快乐的, 所以 2 号女生也变得快乐了;

第 3 天, 1 号男生和 0 号女生吃饭, 因为 0 号女生是快乐的, 所以 1 号男生也变得快乐了;

第 4 天, 0 号男生和 1 号女生吃饭, 因为 0 号男生是快乐的, 所以 1 号女生也变得快乐了;

这样就全都快乐了

对于第二组数据: 1 号男生, 1 号女生和 3 号女生永远也没法变得快乐

数据范围及约定

对于 50% 的数据, $n, m \leq 100, k = 0$

对于 70% 的数据, $n, m, k \leq 2000$

另外存在 10% 的数据, $n, m \leq 1000000000, b, g \leq 100000, k = 0$

对于 100% 的数据, $n, m, k \leq 1000000000, b, g, t \leq 100000$

嫌疑人 suspect

问题描述

最近公司里的代码中发现了一个极其严重的 bug，小组长 David 决定找出这个嫌疑人并惩罚他！

为了找出嫌疑人，David 开了一个会。会上有 n 个程序员，每个程序员都发表了如下言论：“不是 x 干的就是 y 干的！”（不同程序员言语中的 x 与 y 不一定相同）

David 决定选出两个嫌疑人，并请他们去他的办公室喝茶。当然 David 的选取方案会参考所有程序员们的建议。他想让他的选取方案至少有 p 个人支持。一个程序员会支持一个方案当且仅当至少有一个该程序员认定的嫌疑人被请去喝茶了。

注意：假如某个程序员被选为两个嫌疑人之一，他甚至可能支持这个选取方案，只要选取的另一个人是他说的两个嫌疑人之一即可

输入格式

第一行两个整数 n, p ，表示程序员的个数和一个选取方案至少需要多少个程序员同意。

接下来 n 行，第 i 行两个整数 $x_i, y_i (x_i \neq i, y_i \neq i, x_i \neq y_i)$ ，表示 i 号程序员认定的两个嫌疑人

输出格式

输出一个整数，表示有多少种合法的选取方案。

注意：选取的两个人的顺序是无所谓的，即：选 $(1, 2)$ 两人和选 $(2, 1)$ 两人是一样的选取方案

样例输入

8 6
5 6
5 7
5 8
6 2
2 1
7 3
1 3
1 4

样例输出

1

样例解释

选 1 和 5 去喝茶，总共会有 6 个人支持

数据范围及约定

存在 10% 的数据， $p = 0$

另外存在 20% 的数据， $p = 1$

另外存在 25% 的数据， $n \leq 100$

另外存在 25% 的数据， $n \leq 1000$

对于 100% 的数据， $n \leq 300000, 0 \leq p \leq n$

异或 xor

问题描述

David 最近正在研究异或运算，异或运算拥有许多美妙的性质。为了更好的理解异或运算，他需要做如下的一个实验：

有一个 n 个元素的数列 a ，要进行 m 次查询，每次查询形式如下：

1. 给出两个整数 l, r ，表示查询区间的左右端点
2. 取出区间 $[l, r]$ 中的所有出现了**偶数次**的整数。比如 $1, 2, 1, 2, 1$ ，则会取出一个数 2
3. 将取出来的数全部异或起来，并将该异或值作为本次查询的答案。形式化来说，设取出的数为 x_1, x_2, \dots, x_n ，则计算 $x_1 \oplus x_2 \oplus \dots \oplus x_n$ ，其中 \oplus 表示异或运算

由于 David 计算能力很差，于是他请你来帮他计算一下每次询问的答案

输入格式

第一行一个整数 n ，表示数列的长度

接下来一行 n 个非负整数，表示 a 数组中的每个元素

接下来一行一个整数 m ，表示查询的数量

接下来 m 行，每行两个整数 l, r 表示这次查询区间的左右端点

输出格式

对于每组查询，输出一行一个整数，表示这组查询的答案

样例输入 1

```
3
3 7 8
1
1 3
```

样例输出 1

```
0
```

样例输入 2

```
7
1 2 1 3 3 2 3
5
4 7
4 5
1 3
1 7
1 5
```

样例输出 2

```
0
3
1
3
2
```


样例解释

对于第二组样例：第一组询问区间中没有出现偶数次的数字，所以输出 0

第二组询问区间中，3 出现了偶数次，所以答案为 3

第三组询问中，1 出现了偶数次，所以答案为 1

第四组询问中，1, 2 出现了偶数次，所以答案为 $1 \oplus 2 = 3$

第五组询问中，1, 3 出现了偶数次，所以答案为 $1 \oplus 3 = 2$

数据范围及约定

存在 10% 的数据，数列中没有两个一样的数

另外存在 25% 的数据， $n, m \leq 1000, a_i \leq 1000$

另外存在 20% 的数据， $n, m \leq 1000$

另外存在 20% 的数据， $0 \leq a_i \leq 1$

对于 100% 的数据， $n, m \leq 300000, 0 \leq a_i \leq 1000000000, 1 \leq l \leq r \leq n$