

# GGF NOIP 模拟题

题目名称	线性筛	断电	多边形
题目类型	传统型	传统型	传统型
目录	linear	endpoint	effect
可执行文件名	linear	endpoint	effect
输入文件名	linear.in	endpoint.in	effect.in
输出文件名	linear.out	endpoint.out	effect.out
每个测试点时限	1.0 秒	5.0 秒	4.0 秒
内存限制	512 MB	512 MB	512 MB
测试点/包数目	10	10	20
测试点是否等分	是	是	是

提交源程序文件名

对于 C++ 语言	linear.cpp	endpoint.cpp	effect.cpp
对于 C 语言	linear.c	endpoint.c	effect.c
对于 Pascal 语言	linear.pas	endpoint.pas	effect.pas

注意事项:

- 1、提交的源文件必须存放在已建立好的下发样例的文件夹中（该文件夹与试题同名）。
- 2、文件名（包括程序名和输入输出文件名）必须使用英文小写。
- 3、结果比较方式为忽略行末空格、文末回车后的全文比较。
- 4、C/C++ 中函数 `main()` 的返回值类型必须是 `int`，值为 0。
- 5、对于因未遵守以上规则对成绩造成的影响，相关申诉不予受理。

毒瘤的出题人 1



dark 的出题人 2



# 线性筛(linear)

## Problem statement

Orange-User 最近在学习线性筛，Yuhuger 前来凑热闹，顺便来嘲讽一下 Orange-User。

Orange-User 看出了 Yuhuger 的意图，于是摆出了一个难题来考验 Yuhuger。

众所周知，线性筛的一种伪代码如下

输入一个整数  $n$

申请一个布尔数组  $b$ ，初始值全为假

申请一个质数列表  $c$ ，初始为空

令  $i$  从 2 到  $n$

    如果  $b[i]$  为假，那么  $c$  在尾部增加  $i$  这个元素

    令  $j$  从头至尾遍历  $c$  的元素

        使  $b[i*j]$  为真

        如果  $i$  为  $j$  的倍数，那么跳出这层循环

缩进表示语句之间的关系，缩进多的语句嵌套在缩进少的语句中。

但是 Orange-User 将线性筛的伪代码变成了这样：

输入一个整数  $n$

申请一个整型数组  $b$ ，初始值全为 0

申请一个质数列表  $c$ ，初始为空

令  $i$  从 2 到  $n$

    如果  $b[i]$  为 0，那么  $c$  在尾部增加  $i$  这个元素

    令  $j$  从头至尾遍历  $c$  的元素

        使  $b[i*j]$  为  $j$

        如果  $i$  为  $j$  的倍数，那么跳出这层循环

现在 Orange-User 给定整数  $l$  和  $r$

Yuhuger 需要回答两个问题：

1: 对于未经 Orange-User 修改的伪代码，输入整数为  $r$ ，执行之后，布尔数组  $b$  中下标在  $[l,r]$  的值为真的个数。

2: 对于 Orange-User 修改后的伪代码，输入整数为  $r$ ，执行之后，整型数组  $b$  中下标在  $[l,r]$  的值的和。

Yuhuger 冷静地思考了一会儿，发现自己并没有什么想法，转而向你求助。你能一眼秒了此题，加入嘲讽 Orange-User 的行列吗？

## Input

输入文件共一行，两个用空格隔开的正整数  $l,r$

## Output

输出文件共一行，两个用空格隔开的整数。第一个整数为第一个问题的答案，第二个整

数为第二个问题的答案对 998244353 取模的结果

### Example Input 1

5 10

### Example output 1

4 9

### Explanation1

第一个问题的答案 4 为:  $b[6], b[8], b[9], b[10]$  为真

第二个问题的答案 9 为:  $b[6]=b[8]=b[10]=2, b[9]=3$

### Example Input 2

24788 87347

### Example Output 2

56820 605038

### Constraints

对于%10 的数据  $1 \leq r \leq 2e7$

对于另%10 的数据  $r \leq l+100$

对于另%30 的数据  $r \leq l+10000$

对于另%30 的数据  $r \leq l+1000000$

对于所有数据  $1 \leq l \leq r \leq 1e12, r \leq l+10000000$

# 断电(endpoint)

## Problem statement

暑假里的一天，Orange-User 正在机房里专注地学习线性筛。

在  $n$  个小时过去后，Orange-User 发现自己意识模糊，于是开始颓 nohtyp 语言突然间机房断电了。

Orange-User 恼羞成怒地开始检查线路。在眼查了  $n$  年之后，Orange-User 决定开始调试。

众所周知，调试需要工具，Orange-User 却找不到任何可以使用的电工工具。机房外有一个电工工具箱，但是上面的密码锁太过复杂，无法破解。

Orange-User 选择相信自己的人品，开始尝试猜出密码。

$n$  年后人品过人的 Orange-User，终于猜出了密码，打开了密码锁，机房早通电  $n$  年了。

Orange-User 后悔了，于是回到了过去，尝试破解密码。

密码破解中的一部分需要一个强大的数据结构：

现在有  $q$  次操作，共有两种：

1. 加入一个数。
2. 查询一个数，输出该数与之前加入的数的最大异或值，最大 and 值，最大 or 值。

Orange-User 对此毫无想法，现在，该你上了。

## Input

输入文件共  $q$  行，第一行为一个正整数  $q$ 。

接下来  $q$  行，每行两个整数  $tp, x$  描述一次操作，如果  $tp$  为 1，加入  $x$ ，如果  $tp$  为 3，查询  $x$ ，只输出最大异或值，如果  $tp$  为 2，查询  $x$ ，只输出三个值。

## Output

输出文件共查询次数行，每行有 1 或 3 个用空格隔开的整数，含义如上。

## Example Input 1

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

## Example output 1

```
7 0 7
5 5 7
```

## Explanation1

询问 4 时，已插入 2、3，最大异或值为  $4^3=7$ ，最大 and 值为  $4\&3$  或  $4\&2=0$ ，最大 or 值为  $4|3=7$

询问 7 时，已插入 2、3、5，最大异或值为  $7^2=5$ ，最大 and 值为  $7\&5=5$ ，最大 or 值为  $7|2=7|3=7|5=7$

## Example Input 2

```
10
1 194570
1 202332
1 802413
2 234800
1 1011194
2 1021030
2 715144
2 720841
1 7684
2 85165
```

## Example Output 2

```
1026909 201744 1032061
879724 984162 1048062
655316 682376 1043962
649621 683464 1048571
926039 85160 1011199
```

## Constraints

对于%10 的数据  $1 \leq q \leq 5000$

对于另%10 的数据保证  $x < 1024$

对于另%20 的数据保证没有操作 2

对于另%40 的数据保证  $q \leq 300000$

对于所有数据保证  $1 \leq q < 2^{20}, 1 \leq x < 2^{20}$  保证第一个操作为 1 操作。

# 多边形(effect)

## Problem Background

其实某国语言中多边形叫 effect(我才不会说是贺了个板子忘改了呢)

## Problem Statement

Yuhuger 在机房停电时也在专心地颓，可是被停电影响，Yuhuger 只好找一点乐子。

Yuhuger 发现 Orange-User 在修电路，出于好奇的心理，Yuhuger 上前观察电路的形态。

电路好长呀！

电路好复杂呀！

电路好难呀！

Yuhuger 连物理作业都不会做怎么修电路呢？

可是搞 OI 的怎么能不会修电路呢！

经过仔细观察，Yuhuger 终于将电路抽象化了。

电路有  $n$  个电路板， $m$  条双向导线，每条导线连接两个电路板，电子可以在导线中自由移动，所以电流可以从导线任意一端流向另一端，而且电流的方向与电子的移动方向相反，总而言之这是个无向图。

每个电路板上有个电路元器件，每个电路元器件有启动电压和功能属性。只有电压大于等于启动电压，才能启动电路元器件。只有一条导线两端电路板上的电路元器件都启动，这条导线才会有电流经过（这同时意味着这两个电路元器件有关联，关联有传递性）。

由于机房断电的复杂性，如果一个极大的两两有关联的电路元器件集合中，某种功能属性的元器件存在且个数符合某个 **dark** 的条件——一个数是某个 **dark** 常数  $kk$  的倍数，那么视为这个元器件集合有这种功能属性。

由于机房断电的复杂性，Orange-User 只能先切断了电源，再给出一定的启动电压，让某些元器件启动。很显然，极大的两两有关联的电路元器件组成了一个集合  $A$ ，所有的可以启动的元器件组成了若干个这样的集合。而一个集合  $A$  的功能值是指  $A$  集合不同功能属性的数量。

为了让功能恢复地尽可能地快，Orange-User 希望知道在这个启动电压下，在所有可以启动的  $A$  中，最大的功能值。

本来这非常简单，可是 Yuhuger 不慎踩坏了一些电路元器件，这不仅使该元器件报废，还使得在该电压下所有与该元器件有关联的所有元器件报废了。

Orange-User 无能为力，只好思考新的问题，Orange-User 希望知道在这个启动电压下，在所有没有报废的  $A$  中，最大的功能值。

## Input

输入文件开始为两个正整数  $n, m, kk$

然后两行，每行  $n$  个整数。

第一行为每个电路板的电路元器件的启动电压  $a[i]$ ，

第二行为每个电路板的电路元器件的功能属性  $b[i]$ 。

然后为  $m$  行，每行两个不相同的整数  $x,y$  表示  $m$  条边的端点(保证无重边，端点合法)  
这之后为一个正整数  $q$ ，表示询问个数。

接下来  $q$  行每行描述一个询问，

每行第一个数为启动电压  $V$ ，第二个数为 Yuhuger 踩坏元器件的个数  $k$ ，接下来  $k$  个数表示踩坏的元器件所在电路板（不保证两两不同）

## Output

输出文件共  $q$  行，每行一个整数，表示该次询问的答案。

若该次询问中没有 没有报废的 A 请输出 0

## Example Input

```
5 4 2
3 7 2 5 4
1 3 3 2 2
1 3
1 2
3 4
3 5
5
4 0
4 1 3
1 0
5 0
10 0
```

## Example Output

```
0
0
0
1
2
```

## Explanation

第一次询问，启动电压为 4，启动了 1、3、5 号点，它们之间有关联。

同时 1、3、2 三种功能属性各有一种，没有 2 的倍数个。

第二次询问，和第一问类似，由于 3 号点所在的集合被踩坏了，没有可以启动的元器件集合。

第三次询问，启动电压太低，启动不了任何点。



第四次询问，启动了 1、3、4、5，功能属性分别为 1、3、2、2，2 功能属性的有 2 个，所以答案为 1。

第五次询问，启动了所有点，功能属性分贝为 1、3、3、2、2，3、2 功能属性各有两个，所以答案为 2。

Constraints

注：“无”表示无特殊性质

测试点编号	n	m	q	$\sum k$
1,2,3,4,5,6	$\leq 100$	$\leq 1000$	$\leq 10000$	无
7,8,9,10	$\leq 1000$	无	无	=0
11,12,13,14,15, 16,17,18,19,20	无	无	无	无

对于所有数据

$$1 \leq n \leq 200000, 1 \leq m \leq \min(n * \frac{n-1}{2}, 500000), 1 \leq q \leq 200000, 1 \leq k \leq n$$
$$\sum k \leq 200000, 1 \leq a[i], b[i] \leq INT\_MAX$$