

# 第 22 届全国青少年信息学奥林匹克联赛

## CCF-NOIP-2016

### 提高组（复赛） 第二试

样例后面的数字表示题目对应的那挡数据

题目名称	组合数问题	蚯蚓	愤怒的小鸟
题目类型	传统型	传统型	传统型
目录	problem	earthworm	angrybirds
可执行文件名	problem	earthworm	angrybirds
输入文件名	problem.in	earthworm.in	angrybirds.in
输出文件名	problem.out	earthworm.out	angrybirds.out
每个测试点时限	1.0 秒	1.0 秒	2.0 秒
内存限制	512 MB	512 MB	512 MB
测试点数目	20	20	20
每个测试点分值	5	5	5

提交源程序文件名

对于 C++ 语言	problem.cpp	earthworm.cpp	angrybirds.cpp
对于 C 语言	problem.c	earthworm.c	angrybirds.c
对于 Pascal 语言	problem.pas	earthworm.pas	angrybirds.pas

编译选项

对于 C++ 语言	-lm	-lm	-lm
对于 C 语言	-lm	-lm	-lm
对于 Pascal 语言			

注意事项：

1. 文件名（程序名和输入输出文件名）必须使用英文小写。
2. 除非特殊说明，结果比较方式均为忽略行末空格及文末回车的全文比较。
3. 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。
4. 全国统一评测时采用的机器配置为：CPU AMD Athlon(tm) II x2 240 processor, 2.8GHz，内存 4G，上述时限以此配置为准。
5. 只提供 Linux 格式附加样例文件。
6. 评测在 NOI Linux 下进行。
7. 编译时不打开任何优化选项。

## 组合数问题（problem）

## 【问题描述】

组合数  $C_n^m$  表示的是从  $n$  个物品中选出  $m$  个物品的方案数。举个例子，从  $(1, 2, 3)$  三个物品中选择两个物品可以有  $(1, 2), (1, 3), (2, 3)$  这三种选择方法。根据组合数的定义，我们可以给出计算组合数  $C_n^m$  的一般公式：

$$C_n^m = \frac{n!}{m!(n-m)!}$$

其中  $n! = 1 \times 2 \times \cdots \times n$ 。

小葱想知道如果给定  $n, m$  和  $k$ ，对于所有的  $0 \leq i \leq n, 0 \leq j \leq \min(i, m)$  有多少对  $(i, j)$  满足  $C_i^j$  是  $k$  的倍数。

## 【输入格式】

从文件 `problem.in` 中读入数据。

第一行有两个整数  $t, k$ ，其中  $t$  代表该测试点总共有多少组测试数据， $k$  的意义见【问题描述】。

接下来  $t$  行每行两个整数  $n, m$ ，其中  $n, m$  的意义见【问题描述】。

## 【输出格式】

输出到文件 `problem.out` 中。

$t$  行，每行一个整数代表所有的  $0 \leq i \leq n, 0 \leq j \leq \min(i, m)$  中有多少对  $(i, j)$  满足  $C_i^j$  是  $k$  的倍数。

## 【样例 1 输入】

```
1 2
3 3
```

## 【样例 1 输出】

```
1
```

## 【样例 1 说明】

在所有可能的情况中，只有  $C_2^1 = 2$  是 2 的倍数。

## 【样例 2 输入】

2 5  
4 5  
6 7

## 【样例 2 输出】

7

## 【子任务】

测试点	$n$	$m$	$k$	$t$
1	$\leq 3$	$\leq 3$	$= 2$	$= 1$
2			$= 3$	$\leq 10^4$
3	$\leq 7$	$\leq 7$	$= 4$	$= 1$
4			$= 5$	$\leq 10^4$
5	$\leq 10$	$\leq 10$	$= 6$	$= 1$
6			$= 7$	$\leq 10^4$
7	$\leq 20$	$\leq 100$	$= 8$	$= 1$
8			$= 9$	$\leq 10^4$
9	$\leq 25$	$\leq 2000$	$= 10$	$= 1$
10			$= 11$	$\leq 10^4$
11	$\leq 60$	$\leq 20$	$= 12$	$= 1$
12			$= 13$	$\leq 10^4$
13	$\leq 100$	$\leq 25$	$= 14$	$= 1$
14			$= 15$	$\leq 10^4$
15		$\leq 60$	$= 16$	$= 1$
16			$= 17$	$\leq 10^4$
17	$\leq 2000$	$\leq 100$	$= 18$	$= 1$
18			$= 19$	$\leq 10^4$
19		$\leq 2000$	$= 20$	$= 1$
20			$= 21$	$\leq 10^4$

## 四点旅行(four)

时间限制: 1.000 Sec 内存限制: 128 MB

### 题目描述

给定一个 $n$ 个点, $m$ 条边的有向图,边的长度都是1.

设 $dis(x, y)$ 为 $x$ 到 $y$ 的最短路径.

求四个不同的点 $a, b, c, d$ ,满足 $dis(a, b) + dis(b, c) + dis(c, d)$ 最大.

输出最大距离.

### 输入

第一行两个整数 $n$ 和 $m$

接下来 $m$ 行,每行两个整数 $a$ 和 $b$ ,表示一条边 $a$ 到 $b$ 的边.

### 输出

输出一个整数表示最大距离.

### 样例输入

```
8 9
1 2
2 3
3 4
4 1
4 5
5 6
6 7
7 8
8 5
```

### 样例输出

```
13
```

### 提示

$dis(2, 1) = 3$ ,  $dis(1, 8) = 7$ ,  $dis(8, 7) = 3$ . 最大长度为 13. 可以重复经过一个点.

数据保证有解.

对于30%的数据,  $n$ 的范围 $[10, 50]$ ,  $m$ 的范围 $[10, 200]$ ;

对于60%的数据,  $n$ 的范围 $[10, 200]$ ,  $m$ 的范围 $[10, 2000]$ ;

对于100%的数据,  $n$ 的范围 $[10, 2000]$ ,  $m$ 的范围 $[10, 5000]$ ;

## 蚯蚓 (earthworm)

## 【问题描述】

本题中，我们将用符号  $\lfloor c \rfloor$  表示对  $c$  向下取整，例如： $\lfloor 3.0 \rfloor = \lfloor 3.1 \rfloor = \lfloor 3.9 \rfloor = 3$ 。

蛐蛐国最近蚯蚓成灾了！隔壁跳蚤国的跳蚤也拿蚯蚓们没办法，蛐蛐国王只好去请神刀手来帮他们消灭蚯蚓。

蛐蛐国里现在共有  $n$  只蚯蚓（ $n$  为正整数）。每只蚯蚓拥有长度，我们设第  $i$  只蚯蚓的长度为  $a_i$  ( $i = 1, 2, \dots, n$ )，并保证所有的长度都是非负整数（即：可能存在长度为 0 的蚯蚓）。

每一秒，神刀手会在所有的蚯蚓中，准确地找到最长的那一只（如有多个则任选一个）将其切成两半。神刀手切开蚯蚓的位置由常数  $p$ （是满足  $0 < p < 1$  的有理数）决定，设这只蚯蚓长度为  $x$ ，神刀手会将其切成两只长度分别为  $\lfloor px \rfloor$  和  $x - \lfloor px \rfloor$  的蚯蚓。特殊地，如果这两个数的其中一个等于 0，则这个长度为 0 的蚯蚓也会被保留。此外，除了刚刚产生的两只新蚯蚓，其余蚯蚓的长度都会增加  $q$ （是一个非负整常数）。

蛐蛐国王知道这样不是长久之计，因为蚯蚓不仅会越来越多，还会越来越长。蛐蛐国王决定求助于一位有着洪荒之力的神秘人物，但是救兵还需要  $m$  秒才能到来……（ $m$  为非负整数）

蛐蛐国王希望知道这  $m$  秒内的战况。具体来说，他希望知道：

- $m$  秒内，每一秒被切断的蚯蚓被切断前的长度（有  $m$  个数）；
- $m$  秒后，所有蚯蚓的长度（有  $n + m$  个数）。

蛐蛐国王当然知道怎么做啦！但是他想考考你……

## 【输入格式】

从文件 *earthworm.in* 中读入数据。

第一行包含六个整数  $n, m, q, u, v, t$ ，其中： $n, m, q$  的意义见【问题描述】； $u, v, t$  均为正整数；你需要自己计算  $p = u/v$ （保证  $0 < u < v$ ）； $t$  是输出参数，其含义将会在【输出格式】中解释。

第二行包含  $n$  个非负整数，为  $a_1, a_2, \dots, a_n$ ，即初始时  $n$  只蚯蚓的长度。

同一行中相邻的两个数之间，恰好用一个空格隔开。

保证  $1 \leq n \leq 10^5$ ， $0 \leq m \leq 7 \times 10^6$ ， $0 < u < v \leq 10^9$ ， $0 \leq q \leq 200$ ， $1 \leq t \leq 71$ ， $0 \leq a_i \leq 10^8$ 。

## 【输出格式】

输出到文件 *earthworm.out*

第一行输出  $\lfloor \frac{m}{t} \rfloor$  个整数，按时间页序，依次输出第  $t$  秒，第  $2t$  秒，第  $3t$  秒，……被切断蚯蚓（在被切断前）的长度。

第二行输出  $\lfloor \frac{n+m}{t} \rfloor$  个整数，输出  $m$  秒后蚯蚓的长度：需要按从大到小的页序，依次输出排名第  $t$ ，第  $2t$ ，第  $3t$ ，..... 的长度。

同一行中相邻的两个数之间，恰好用一个空格隔开。即使某一行没有任何数需要输出，你也应输出一个空行。

请阅读样例来更好地理解这个格式。

**【样例 1 输入】**

```
3 7 1 1 3 1
3 3 2
```

**【样例 1 输出】**

```
3 4 4 4 5 5 6
6 6 6 5 5 4 4 3 2 2
```

**【样例 1 说明】**

在神刀手到来前：3 只蚯蚓的长度为 3,3,2。

1 秒后：一只长度为 3 的蚯蚓被切成了两只长度分别为 1 和 2 的蚯蚓，其余蚯蚓的长度增加了 1。最终 4 只蚯蚓的长度分别为(1,2),4,3。括号表示这个位置刚刚有一只蚯蚓被切断。

2 秒后：一只长度为 4 的蚯蚓被切成了 1 和 3。5 只蚯蚓的长度分别为：2,3,(1,3),4。

3 秒后：一只长度为 4 的蚯蚓被切断。6 只蚯蚓的长度分别为：3,4,2,4,(1,3)。

4 秒后：一只长度为 4 的蚯蚓被切断。7 只蚯蚓的长度分别为：4,(1,3),3,5,2,4。

5 秒后：一只长度为 5 的蚯蚓被切断。8 只蚯蚓的长度分别为：5,2,4,4,(1,4),3,5。

6 秒后：一只长度为 5 的蚯蚓被切断。9 只蚯蚓的长度分别为：(1,4),3,5,5,2,5,4,6。

7 秒后：一只长度为 6 的蚯蚓被切断。10 只蚯蚓的长度分别为：2,5,4,6,6,3,6,5,(2,4)。

所以，7 秒内被切断的蚯蚓的长度依次为 3,4,4,4,5,5,6。7 秒后，所有蚯蚓长度从大到小排序为 6,6,6,5,5,4,4,3,2,2。

**【样例 2 输入】**

```
3 7 1 1 3 2
3 3 2
```

**【样例 2 输出】**

```
4 4 5
6 5 4 3 2
```

**【样例 2 说明】**

这个数据中只有  $t=2$  与上个数据不同。只需在每行都改为每两个数输出一个数即可。

虽然第一行最后有一个 6 没有被输出，但是第二行仍然要重新从第二个数再开始输出。

**【样例 3 输入】**

```
3 7 1 1 3 9
3 3 2
```

**【样例 3 输出】**

```
2
```

**【样例 3 说明】**

这个数据中只有  $t=9$  与上个数据不同。

注意第一行没有数要输出，但也要输出一个空行。

**【子任务】**

- 测试点 1~3 满足  $m=0$ 。
  - 测试点 4~7 满足  $n, m \leq 1,000$ 。
  - 测试点 8~14 满足  $q=0$ ，其中测试点 8~9 还满足  $m \leq 10^5$ 。
  - 测试点 15~18 满足  $m \leq 3 \times 10^5$ 。
  - 测试点 19~20 没有特殊的约定，参见原始的数据范围。
  - 测试点 1~12, 15~16 还满足  $v \leq 2$ ，这意味着  $u, v$  的唯一可能的取值是  $u=1, v=2$ ，即  $p=0.5$ 。这可能会对解决问题有特殊的帮助。
- 每个测试点的详细数据范围见下表。

测试点	$n$	$m$	$t$	$a_i$	$v$	$q$
1	$= 1$	$= 0$	$= 1$	$\leq 10^6$	$\leq 2$	$= 0$
2	$= 10^3$					
3	$= 10^5$					
4	$= 1$	$= 10^3$				$\leq 200$
5	$= 10^3$					
6	$= 1$					
7	$= 10^3$					
8	$= 5 \times 10^4$	$= 5 \times 10^4$				$= 0$
9	$= 10^5$	$= 10^5$	$= 2$			
10		$= 2 \times 10^6$	$= 21$			
11		$= 2.5 \times 10^6$	$= 26$			
12		$= 3.5 \times 10^6$	$= 36$			
13		$= 5 \times 10^6$	$= 51$			
14		$= 7 \times 10^6$	$= 71$			
15	$= 5 \times 10^4$	$= 5 \times 10^4$	$= 1$	$\leq 10^7$	$\leq 10^9$	$= 0$
16		$= 1.5 \times 10^5$	$= 2$			
17	$= 10^5$	$= 10^5$	$= 3$	$\leq 10^8$	$\leq 10^9$	$\leq 200$
18		$= 3 \times 10^5$	$= 4$			
19		$= 3.5 \times 10^6$	$= 36$			
20		$= 7 \times 10^6$	$= 71$			



## 愤怒的小鸟（angrybirds）

### 【问题描述】

Kiana 最近沉迷于一款神奇的游戏无法自拔。

简单来说，这款游戏是在一个平面上进行的。

有一架弹弓位于  $(0, 0)$  处，每次 Kiana 可以用它向第一象限发射一只红色的小鸟，小鸟们的飞行轨迹均为形如  $y = ax^2 + bx$  的曲线，其中  $a, b$  是 Kiana 指定的参数，且必须满足  $a < 0$ 。

当小鸟落回地面（即  $x$  轴）时，它就会瞬间消失。

在游戏的某个关卡里，平面的第一象限中有  $n$  只绿色的小猪，其中第  $i$  只小猪所在的坐标为  $(x_i, y_i)$ 。

如果某只小鸟的飞行轨迹经过了  $(x_i, y_i)$ ，那么第  $i$  只小猪就会被消灭掉，同时小鸟将会沿着原先的轨迹继续飞行；

如果一只小鸟的飞行轨迹没有经过  $(x_i, y_i)$ ，那么这只小鸟飞行的全过程就不会对第  $i$  只小猪产生任何影响。

例如，若两只小猪分别位于  $(1, 3)$  和  $(3, 3)$ ，Kiana 可以选择发射一只飞行轨迹为  $y = -x^2 + 4x$  的小鸟，这样两只小猪就会被这只小鸟一起消灭。而这个游戏的目的，就是通过发射小鸟消灭所有的小猪。这款神奇游戏的每个关卡对 Kiana 来说都很难，所以 Kiana 还输入了一些神秘的指令，使得自己能更轻松地完成这个游戏。这些指令将在【输入格式】中详述。

假设这款游戏一共有  $T$  个关卡，现在 Kiana 想知道，对于每一个关卡，至少需要发射多少只小鸟才能消灭所有的小猪。由于她不会算，所以希望由你告诉她。

### 【输入格式】

从文件 *angrybirds.in* 中读入数据。

第一行包含一个正整数  $T$ ，表示游戏的关卡总数。

下面依次输入这  $T$  个关卡的信息。每个关卡第一行包含两个非负整数  $n, m$ ，分别表示该关卡中的小猪数量和 Kiana 输入的神秘指令类型。接下来的  $n$  行中，第  $i$  行包含两个正实数  $x_i, y_i$ ，表示第  $i$  只小猪坐标为  $(x_i, y_i)$ 。数据保证同一个关卡中不存在两只坐标完全相同的小猪。

如果  $m = 0$ ，表示 Kiana 输入了一个没有任何作用的指令。

如果  $m = 1$ ，则这个关卡将会满足：至多用  $\lceil n/3 + 1 \rceil$  只小鸟即可消灭所有小猪。

如果  $m = 2$ ，则这个关卡将会满足：一定存在一种最优解，其中有一只小鸟消灭了至少  $\lfloor n/3 \rfloor$  只小猪。

保证  $1 \leq n \leq 18$ ， $0 \leq m \leq 2$ ， $0 < x_i, y_i < 10$ ，输入中的实数均保留到小数点后两位。

上文中，符号  $\lceil c \rceil$  和  $\lfloor c \rfloor$  分别表示对  $c$  向上取整和向下取整，例如： $\lceil 2.1 \rceil = \lceil 2.9 \rceil = \lceil 3.0 \rceil = \lfloor 3.0 \rfloor = \lfloor 3.1 \rfloor = \lfloor 3.9 \rfloor = 3$ 。

### 【输出格式】

输出到文件 angrybirds.out 中。

对每个关卡依次输出一行答案。

输出的每一行包含一个正整数，表示相应的关卡中，消灭所有小猪最少需要的小鸟数量。

### 【样例 1 输入】

```
2
2 0
1.00 3.00
3.00 3.00
5 2
1.00 5.00
2.00 8.00
3.00 9.00
4.00 8.00
5.00 5.00
```

### 【样例 1 输出】

```
1
1
```

### 【样例 1 说明】

这组数据中一共有两个关卡。

第一个关卡与【问题描述】中的情形相同，2 只小猪分别位于 (1.00, 3.00) 和 (3.00, 3.00)，只需发射一只飞行轨迹为  $y = -x^2 + 4x$  的小鸟即可消灭它们。

第二个关卡中有 5 只小猪，但经过观察我们可以发现它们的坐标都在抛物线  $y = -x^2 + 6x$  上，故 Kiana 只需要发射一只小鸟即可消灭所有小猪。

**【样例 2 输入】**

```
3
2 0
1.41 2.00
1.73 3.00
3 0
1.11 1.41
2.34 1.79
2.98 1.49
5 0
2.72 2.72
2.72 3.14
3.14 2.72
3.14 3.14
5.00 5.00
```

**【样例 2 输出】**

```
2
2
3
```

**【样例 3 输入】**

```
1
10 0
7.16 6.28
2.02 0.38
8.33 7.78
7.68 2.09
7.46 7.86
5.77 7.44
8.24 6.72
4.42 5.11
5.42 7.79
8.15 4.99
```

## 【样例 3 输出】

6

## 【子任务】

数据的一些特殊规定如下表：

测试点编号	$n$	$m$	$T$
1	$\leq 2$	$= 0$	$\leq 10$
2			$\leq 30$
3	$\leq 3$		$\leq 10$
4			$\leq 30$
5	$\leq 4$		$\leq 10$
6			$\leq 30$
7	$\leq 5$		$\leq 10$
8	$\leq 6$		
9	$\leq 7$		
10	$\leq 8$		
11	$\leq 9$		$\leq 30$
12	$\leq 10$		
13	$\leq 12$	$= 1$	
14		$= 2$	
15	$\leq 15$	$= 0$	$\leq 15$
16		$= 1$	
17		$= 2$	
18	$\leq 18$	$= 0$	$\leq 5$
19		$= 1$	
20		$= 2$	