

第 40 届全国信息学奥林匹克竞赛

CCF NOI2023 模拟题

第二试

竞赛时间：7:30–12:20

(请选手务必仔细阅读本页内容)

一. 题目概况

中文题目名称	网格	循环之美	小园丁与老司机
英文题目与子目录名	gird	cyclic	farm
可执行文件名	gird	cyclic	farm
输入文件名	gird.in	cyclic.in	farm.in
输出文件名	gird.out	cyclic.out	farm.out
每个测试点时限	2.0 秒	2.0 秒	1.0 秒
测试点数目	25	25	20
测试点是否等分	是	是	是
附加样例文件	有	有	有
结果比较方式	全文比较 (过滤行末空格及文末回车)		
题目类型	传统型	传统型	传统型
运行内存上限	1024MB	512MB	512MB

二. 提交源程序文件名

对于 C++ 语言	gird.cpp	cyclic.cpp	farm.cpp
-----------	----------	------------	----------

三. 编译选项

对于 C++ 语言	-lm -std=c++14 -O2
-----------	--------------------

四. 注意事项:

- 文件名 (程序名和输入输出文件名) 必须使用英文小写。
- 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`。
- 全国统一评测时采用的机器配置为: Intel(R) Core(TM) i7-8700K CPU @ 3.70GHz, 内存 8GB。上述时限以此配置为准。
- 只提供 Linux 格式附加样例文件。
- 特别提醒: 评测在当前最新公布的 NOI Linux 下进行, 各语言的编译器版本以其为准。

1. 网格(gird)

【问题描述】



跳蚤国王和蚰蚰国王在玩一个游戏。

他们在一个 n 行 m 列的网格上排兵布阵。其中的 c 个格子中 ($0 \leq c \leq nm$), 每个格子有一只蚰蚰, 其余的格子中, 每个格子有一只跳蚤。

我们称占据的格子有公共边的两只跳蚤是**相邻**的。

我们称两只跳蚤是**连通**的, 当且仅当这两只跳蚤**相邻**, 或存在另一只跳蚤与这两只跳蚤都**连通**。

现在, 蚰蚰国王希望, 将某些 (0 个, 1 个或多个) 跳蚤替换成蚰蚰, 使得在此之后存在至少两只跳蚤**不连通**。

例如: 我们用图表示一只跳蚤, 用图表示一只蚰蚰, 那么图 1 描述了一个 $n = 4, m = 4, c = 2$ 的情况。

这种情况下蚰蚰国王可以通过将第 2 行第 2 列, 和第 3 行第 3 列的两只跳蚤替换为蚰蚰, 从而达成他的希望, 如图 2 所示。并且, 不存在更优的方案, 但是可能存在其他替换 2 只跳蚤的方案。

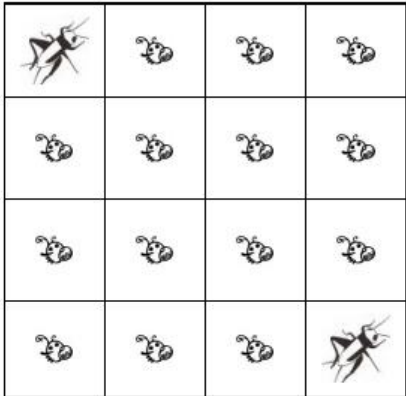


图 1

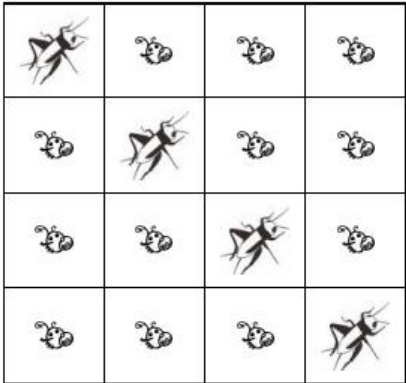


图 2

你需要首先判断蚰蚰国王的希望能否被达成。如果能够达成, 你还需要最小化被替换的跳蚤的个数。

【输入格式】

输入文件为 *grid.in*。

每个输入文件包含多组数据。

输入文件的第一行只有一个整数 T ，表示数据的组数。保证 $1 \leq T \leq 20$ 。

接下来依次输入 T 组数据，每组数据的第一行包含三个整数 n, m, c 。保证 $1 \leq n, m \leq 10^9$ ， $0 \leq c \leq \min(nm, 10^5)$ 。

接下来 c 行，每行包含两个整数 x, y ，表示第 x 行，第 y 列的格子被一个蚰蚰占据 ($1 \leq x \leq n, 1 \leq y \leq m$)。每一组数据当中，同一个蚰蚰不会被多次描述。

同一行相邻的整数之间由一个空格隔开。

【输出格式】

输出文件为 *grid.out*。

对于每一组数据依次输出一行答案。

如果这组数据中，蚰蚰国王的希望不能被达成，输出 -1 。否则，输出被替换的跳蚤的个数的最小值。

【样例 1 输入】

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

【样例 1 输出】

```
2
1
0
-1
```

【样例 1 说明】

第一组数据就是问题描述中的例子。

对于第二组数据，可以将第 2 行第 2 列的一只跳蚤替换为蚰蚰，从而使得存在两只跳蚤**不连通**，并且不存在更优的方案。

对于第三组数据，最初已经存在两只跳蚤**不连通**，故不需要再进行替换。

对于第四组数据，由于最多只有一只跳蚤，所以无论如何替换都不能存在两只跳蚤**不连通**。

【样例 2 输入输出】

见选手目录下的 *grid/grid2.in* 与 *grid/grid2.ans*。

【提示】

如果你的程序需要用到较大的栈空间（这通常意味着需要较深层数的递归），请务必仔细阅读选手目录下的文档 *stack.pdf*，以了解在最终评测时栈空间的限制与在当前工作环境下调整栈空间限制的方法。

【子任务】

对于全部的测试点，保证 $1 \leq T \leq 20$ 。我们记 $\sum c$ 为某个测试点中，其 T 组输入数据的所有 c 的总和。对于所有的测试点， $\sum c \leq 10^5$ 。

对于全部的数据，满足 $1 \leq n, m \leq 10^9, 0 \leq c \leq nm, 1 \leq x \leq n, 1 \leq y \leq m$ 。

每个测试点的详细数据范围见下表。表中的 n, m, c 均是对于单个输入数据（而非测试点）而言的，也就是说同一个测试点下的 T 组数据均满足限制条件；而 $\sum c$ 是对于单个测试点而言的。为了方便阅读，“测试点”一列被放到了表格的中间而不是左边。

n, m		测试点	c
$nm \leq 4$		1	$c \leq nm$
$nm \leq 8$		2	
$nm \leq 15$		3	
$nm \leq 30$		4	
$nm \leq 100$		5	
$nm \leq 300$		6	
$nm \leq 1000$		7	
$nm \leq 20000$		8	$c \leq 5$
		9	$c \leq 15$
		10	$c \leq 30$
$n, m \leq 20000$	$nm \leq 20000$	11	$\sum c \leq 20000$
	$nm \leq 10^5$	12	
	$nm \leq 3 \times 10^5$	13	
	$nm \leq 10^6$	14	
	$nm \leq 10^9$	15	
$n, m \leq 10^5$		16	$\sum c \leq 10^5$
$n, m \leq 10^9$		17	$c = 0$
		18	$c \leq 1$
		19	$c \leq 2$
		20	$c \leq 3$
		21	$c \leq 10$
		22	$c \leq 30$
		23	$c \leq 300$
		24	$\sum c \leq 20000$
		25	$\sum c \leq 10^5$

2. 循环之美(cyclic)

【问题描述】

牛牛是一个热爱算法设计的高中生。在他设计的算法中，常常会使用带小数的数进行计算。牛牛认为，如果在 k 进制下，一个数的小数部分是纯循环的，那么它就是美的。

现在，牛牛想知道：对于已知的十进制数 n 和 m ，在 k 进制下，有多少个数值上互不相等的纯循环小数，可以用分数 $\frac{x}{y}$ 表示，其中 $1 \leq x \leq n$ ， $1 \leq y \leq m$ ，且 x, y 是整数。

一个数是纯循环的，当且仅当其可以写成以下形式：

$$a.\dot{c}_1c_2c_3\dots c_{p-1}\dot{c}_p$$

其中， a 是一个整数， $p \geq 1$ ；对于 $1 \leq i \leq p$ ， c_i 是 k 进制下的一位数字。

例如，在十进制下， $0.45454545\dots\dots = 0.\dot{4}\dot{5}$ 是纯循环的，它可以用 $\frac{5}{11}$ 、 $\frac{10}{22}$ 等分数表示；在十进制下， $0.1666666\dots\dots = 0.1\dot{6}$ 则不是纯循环的，它可以用 $\frac{1}{6}$ 等分数表示。

需要特别注意的是，我们认为一个整数是纯循环的，因为它的小数部分可以表示成 0 的循环或是 $k-1$ 的循环；而一个小数部分非零的有限小数不是纯循环的。

【输入格式】

输入文件为 `cyclic.in`。

输入文件只有一行，包含三个十进制正整数 n, m, k ，意义如题所述。

【输出格式】

输出文件为 `cyclic.out`。

只输出一行一个整数，表示满足条件的的美的数的个数。

【样例 1 输入】

2 6 10

【样例 1 输出】

4

【样例 1 说明】

满足条件的数分别是：

$$1/1 = 1.0000 \dots$$

$$1/3 = 0.3333 \dots$$

$$2/1 = 2.0000 \dots$$

$$2/3 = 0.6666 \dots$$

1/1 和 2/2 虽然都是纯循环小数，但因为它们相等，因此只计数一次；同样，1/3 和 2/6 也只计数一次。

【样例 2 输入】

23333 666666 310

【样例 2 输出】

5089564081

【提示】

这部分将提供一个将分数化为对应的小数的方法，如果你已经熟悉这个方法，你不必阅读本提示。

分数可以通过除法，用分子除以分母化为对应的小数。有些分数在除法过程中无法除尽，这样的分数在不断进行的除法过程中**余数**一定会重复出现。从商数的个位所对应的余数起，设第一次重复出现的余数前两次出现的位置所对应的商数位分别是小数点后第 a 位和小数点后第 b 位（特殊地：如果其中一个对应的商数位是个位，则认为 $a = 0$ ；不妨设 $a < b$ ），则其循环部分可以用小数点后第 $a + 1$ 位到小数点后第 b 位的循环来表示。

例如：在十进制下，将 $\frac{5}{11}$ 转化为小数时，个位开始的商数依次为 4,5,4,...，对应的余数分别为 6,5,6,...。余数第一次重复出现的位置是个位和小数点后第 2 位，那么 $a = 0, b = 2$ 即其循环部分可以用小数点第 1 位到第 3 位来表示。

表示为： $\frac{5}{11} = 0.45454545 \dots = 0.4\dot{5}$ 。

在十进制下，将 $\frac{1}{6}$ 转化为小数时，个位开始的商数依次为 1,6,6, ..., 对应的余数分别为 4,4,4 余数第一次重复出现的位置是小数点后第 1 位和小数点后第 2 位，即其循环部分可以用小数点后第 2 位来表示。表示为： $\frac{1}{6} = 0.1666 \dots = 0.1\dot{6}$ 。

需要注意的是：**商数**重复出现并不代表进入了循环节。

【子任务】

对于所有的测试点，保证 $1 \leq n \leq 10^9, 1 \leq m \leq 10^9, 2 \leq k \leq 2000$ 。对于每个测试点，有以下约束（其中留空的表示没有特殊的约束）：

测试点 编号	n	m	k	测试点 编号	n	m	k
1	≤ 10	≤ 20	$= 2$	13	$\leq 10^5$	$\leq 10^8$	≤ 100
2	≤ 100	$\leq 10^4$	$= 2$	14	$\leq 2 \times 10^5$		≤ 1000
3	≤ 1000		$= 2$	15	$\leq 5 \times 10^5$		
4	≤ 10000		$= 2$	16	$\leq 10^6$	$\leq 10^8$	≤ 100
5	≤ 10	≤ 20	$= 3$	17	$\leq 2 \times 10^6$		≤ 1000
6	≤ 100	$\leq 10^4$	$= 3$	18	$\leq 5 \times 10^6$		
7	≤ 1000		$= 3$	19	$\leq 10^7$	$\leq 10^8$	≤ 100
8	≤ 10000		$= 3$	20	$\leq 2 \times 10^7$		≤ 1000
9	≤ 10	≤ 20	≤ 100	21	$\leq 2 \times 10^7$		
10	≤ 100	$\leq 10^4$	≤ 100	22	$\leq 10^8$	$\leq 10^8$	
11	≤ 1000		≤ 1000	23	$\leq 10^8$	$\leq 10^8$	
12	≤ 10000			24、25			

3. 小园丁与老司机 (farm)

【问题描述】

小园丁 Mr. S 负责看管一片田野, 田野可以看作一个二维平面。田野上有 n 棵许愿树, 编号 $1, 2, 3, \dots, n$, 每棵树可以看作平面上的一个点, 其中第 i 棵树 ($1 \leq i \leq n$) 位于坐标 (x_i, y_i) 。任意两棵树的坐标均不相同。

老司机 Mr. P 从原点 $(0, 0)$ 驾车出发, 进行若干轮行动。每一轮, Mr. P 首先选择任意一个满足以下条件的方向:

1. 为左、右、上、左上 45° 、右上 45° 五个方向之一。
2. 沿此方向前进可以到达一棵他尚未许愿过的树。

完成选择后, Mr. P 沿该方向直线前进, 必须到达该方向上距离最近的尚未许愿的树, 在树下许愿并继续下一轮行动。如果没有满足条件的方向可供选择, 则停止行动。他会采取最优策略, 在尽可能多的树下许愿。若最优策略不唯一, 可以选择任意一种。

不幸的是, 小园丁 Mr. S 发现由于田野土质松软, 老司机 Mr. P 的小汽车在每轮行进过程中, 都会在田野上留下一条车辙印, 一条车辙印可看作以两棵树(或原点和一棵树)为端点的一条线段。

在 Mr. P 之后, 还有很多许愿者计划驾车来田野许愿, 这些许愿者都会像 Mr. P 一样任选一种最优策略行动。Mr. S 认为非左右方向(即上、左上 45° 、右上 45° 三个方向)的车辙印很不美观, 为了维护田野的形象, 他打算租用一些轧路机, 在这群许愿者到来之前夯实所有“可能留下非左右方向车辙印”的地面。

“可能留下非左右方向车辙印”的地面应当是田野上的若干条线段, 其中每条线段都包含在某一种最优策略的行进路线中。每台轧路机都采取满足以下三个条件

的工作模式:

1. 从原点或任意一棵树出发。
2. 只能向上、左上 45° 、右上 45° 三个方向之一移动, 并且只能在树下改变方向或停止。
3. 只能经过“可能留下非左右方向车辙印”的地面, 但是同一块地面可以被多台轧路机经过。

现在 Mr. P 和 Mr. S 分别向你提出了一个问题:

1. 请给 Mr. P 指出任意一条最优路线。
2. 请告诉 Mr. S 最少需要租用多少台轧路机。

【输入格式】

从文件 *farm.in* 中读入数据。

输入文件的第 1 行包含 1 个正整数 n , 表示许愿树的数量。

接下来 n 行, 第 $i + 1$ 行包含 2 个整数 x_i, y_i , 中间用单个空格隔开, 表示第 i 棵许愿树的坐标。

【输出格式】

输出到文件 *farm.out* 中。

输出文件包括 3 行。

输出文件的第 1 行输出 1 个整数 m ，表示 Mr. P 最多能在多少棵树下许愿。

输出文件的第 2 行输出 m 个整数，相邻整数之间用单个空格隔开，表示 Mr. P 应该依次在哪些树下许愿。

输出文件的第 3 行输出 1 个整数，表示 Mr. S 最少需要租用多少台轧路机。

【样例输入 1】

```
6
-1 1
1 1
-2 2
0 8
0 9
0 10
```

【样例输出 1】

```
3
2 1 3
3
```

【样例说明 1】

最优路线 2 条可许愿 3 次： $(0,0) \rightarrow (1,1) \rightarrow (-1,1) \rightarrow (-2,2)$ 或 $(0,0) \rightarrow (0,8) \rightarrow (0,9) \rightarrow (0,10)$ 。

至少 3 台轧路机，路线是 $(0,0) \rightarrow (1,1)$ ， $(-1,1) \rightarrow (-2,2)$ 和 $(0,0) \rightarrow (0,8) \rightarrow (0,9) \rightarrow (0,10)$ 。

【样例输入 2】

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

【样例输出 2】

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


【样例说明 2】

最优路线唯一： $(0,0) \rightarrow (0,1) \rightarrow (-2,1) \rightarrow (2,1) \rightarrow (3,2)$ ，可许愿 4 次。其中在 $(0,1)$ 许愿后，从 $(-2,1)$ 出发沿着向右的方向能够到达的最近的未许愿过的树是 $(2,1)$ ，所以可以到达 $(2,1)$ 。

而如果沿着 $(0,0) \rightarrow (0,1) \rightarrow (2,1) \rightarrow (-2,1)$ 的方向前进，此时 $(-2,1)$ 右边所有树都是许愿过的，根据题目条件规定，停止前进。故无法获得最优解。

$(0,0) \rightarrow (0,1)$ 与 $(2,1) \rightarrow (3,2)$ 会留下非左右方向车辙印，需 2 台轧路机。

【样例输入输出 3】

见选手目录下的 *farm/farm.in* 与 *farm/farm.ans*。

【数据规模与约定】

所有测试数据的范围和特点如下表所示

测试点编号	n 的规模	x_i, y_i 的规模	备注
1	$n = 5$	$ x_i \leq 100$	
2	$n = 10$	$0 < y_i \leq 100$	
3	$n = 100$	$ x_i \leq 10,000$	
4	$n = 1,000$	$0 < y_i \leq 10,000$	
5	$n = 5,000$	$ x_i \leq 1,000,000$ $0 < y_i \leq 1,000,000$	保证最优路线唯一
6			
7			
8	$n = 5,000$	$ x_i \leq 1,000,000$ $0 < y_i \leq 1,000,000$	保证 y_i 互不相同
9	$n = 50,000$		
10			
11	$n = 5,000$	$ x_i \leq 1,000,000$ $0 < y_i \leq 1,000,000$	保证对于任意整数 Y , 满足 $y_i = Y$ 的树不超过 1,000 棵 存在一种最优解, 使得 轧路机不重复经过同一路面
12			
13	$n = 50,000$		
14			
15	$n = 10,000$	$ x_i \leq 1,000,000,000$ $0 < y_i \leq 1,000,000,000$	保证对于任意整数 Y , 满足 $y_i = Y$ 的树不超过 1,000 棵
16			
17	$n = 30,000$		
18			
19	$n = 50,000$		
20			

【评分方式】

对于每个测试点：

若输出文件的第 1 行正确，得到该测试点 20% 的分数；

若输出文件的前两行正确，得到该测试点 40% 的分数；

若输出文件完全正确，得到该测试点 100% 的分数。