
NOI2012 全国青少年信息学奥林匹克竞赛

四川代表队选拔赛

Day 2

题目名称	奇怪的游戏	Blinker 的仰慕者	Blinker 的噩梦
英文代号	game	admire	nightmare
时限	4 秒	1 秒	5 秒
输入文件	game.in	admire.in	nightmare.in
输出文件	game.out	admire.out	nightmare.out
内存限制	128M	128M	512M
测试点个数	10	10	10
总分	100	100	100

时间：2012 年 4 月 28 日

奇怪的游戏

【题目描述】

Blinker 最近喜欢上一个奇怪的游戏。

这个游戏在一个 $N \times M$ 的棋盘上玩，每个格子有一个数。每次 Blinker 会选择两个相邻的格子，并使这两个数都加上 1。

现在 Blinker 想知道最少多少次能使棋盘上的数都变成同一个数，如果永远不能变成同一个数则输出-1。

【输入】

输入的第一行是一个整数 T ，表示输入数据有 T 轮游戏组成。

每轮游戏的第一行有两个整数 N 和 M ，分别代表棋盘的行数和列数。

接下来有 N 行，每行 M 个数。

【输出】

对于每个游戏输出最少能使游戏结束的次数，如果永远不能变成同一个数则输出-1。

【样例输入】

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

【样例输出】

```
2
-1
```

【数据范围】

对于 30% 的数据，保证 $T \leq 10$ ， $1 \leq N, M \leq 8$

对于 100% 的数据，保证 $T \leq 10$ ， $1 \leq N, M \leq 40$ ，所有数为正整数且小于 1000000000

Blinker 的仰慕者

【题目描述】

Blinker 有非常多的仰慕者，他给每个仰慕者一个正整数编号。而且这些编号还隐藏着特殊的意义，即编号的各位数字之积表示这名仰慕者对 Blinker 的重要度。

现在 Blinker 想知道编号介于某两个值 A,B 之间，且重要度为某个定值 K 的仰慕者编号和。

【输入描述】

输入的第一行是一个整数 N，表示 Blinker 想知道的信息个数。

接下来的 N 行，每行有三个数，A,B,K。表示 Blinker 想知道编号介于 A 和 B 之间的，重要度为 K 的仰慕者的编号和。

【输出描述】

输出 N 行，每行输出介于 A 和 B 之间，重要度为 K 的仰慕者编号和。结果可能很大，模上 20120427。

【输入样例】

```
3
1 14 4
1 30 4
10 60 5
```

【输出样例】

```
18
40
66
```

【样例解释】

第一组样例中，在 1 到 14 之间各位数字之积等于 4 的有 4 和 14，故编号和为 18。

【数据范围】

对于 20% 的数据，保证： $2 \leq A \leq B \leq 1000000000$ ， $1 \leq N \leq 30$

对于 50% 的数据，保证： $2 \leq A \leq B \leq 1000000000000000000$ ， $1 \leq N \leq 30$

对于 100% 的数据，保证： $2 \leq A \leq B \leq 1000000000000000000$ ， $1 \leq N \leq 5000$

Blinker 的噩梦

【题目描述】

一天 Blinker 醒来,发现自己成为了一个二维世界的点,而且被标记上了一个奇怪的值。

这个世界是由 N 个边界互不相交(且不相切)的图形组成,这里图形仅包括圆和凸多边形。每个图形还有一个权值。每次 Blinker 走进或走出某个图形时(相切时经过不算),Blinker 的标记值就会被异或上那个值。

现在,我们记录了 Blinker 在这个世界的 M 天的信息。每天可能发生两件事情,一种是某个图形的权值更改为某个值;另一种是 Blinker 从某个点走到另一个点。

我们假设 Blinker 首次出发前的标记值为 0,我们希望知道他每次到达目的地后的标记值。

【输入】

输入的第一行包含 2 个数, N 和 M , 分别表示这个世界的图形数和记录的天数。

接下来有 N 行, 每行表示一个图形。

如果一行以字符 C 开头, 表示这个图形是一个圆, 后面紧跟着三个实数 x, y, r 和一个整数 v , 分别表示圆的 x 坐标, y 坐标和圆的半径以及该图形对应的值。

如果一行以字符 P 开头, 表示这个图形是凸多边形, 后面紧跟着一个整数 L , 表示凸多边形的点数, 然后后面有 L 对实数 $x_0, y_0, x_1, y_1 \dots$, 表示 L 个点的坐标, 这一行最后一个数是一个整数 v , 表示这个图形对应的值, 保证凸多边形上的点按照顺时针给出。

接下来有 M 行, 每行表示一天的记录信息。

如果一行以字符 Q 开头, 表示这一天 Blinker 出行了, 接下来有 x_0, y_0, x_1, y_1 四个实数, 分别表示出发点的坐标和目的地的坐标。

如果一行以字符 C 开头, 表示这一天某个图形的值改变了, 接下来有两个 i 和 v , 表示输入中第 i 个出现的图形的值变成 v 。

【输出】

对于 Blinker 的每个出行输出他到达目的地后的标记值, 很显然这个值与 Blinker 的路径无关。

【样例输入】

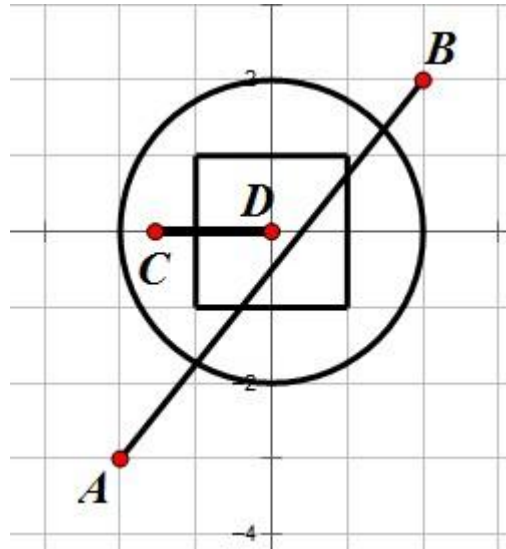
```
2 4
C 0 0 2 1
P 4 -1 -1 -1 1 1 1 1 -1 2
```

```
Q -2 -2 2 2
Q -1.5 0 0.0 0.0
C 1 1005
Q -1.5 0 0.0 0.0
```

【样例输出】

```
0
2
0
```

【样例解释】



样例的世界形如上图：

第一天 Blinker 的初始标记值为 0，从可能从 A 沿直线走到 B，或者他绕过圆走到 B，他的标记值最终都保持不变为 0（假如沿直线从 A 走到 B，共穿过 4 次边界，Blinker 的标记值变化过程为 1,3,1,0）；

第二天 Blinker 的初始标记值为 0，他通过某种不经过图形边界的方法到达了 C 点（即 blink，瞬间移动或闪烁），然后从 C 沿某种路径走到 D，这时他的标记值变为 2；

第三天圆的权值变为 1005；

第四天 Blinker 的初始标记值为 2，他再次回到 C，并再次从 C 走到 D，这时他的标记值又变回 0。

【数据范围】

对于 30% 的数据，保证：

$1 \leq M \leq 10.00$ 凸多边形的点数加上圆的个数小于等于 1000

剩余数据中，保证：

存在一组无图形值变动的数据；

存在一组无凸多边形的数据；

对于 100% 的数据，保证：

$1 \leq N \leq 100000$ ， $1 \leq M \leq 100000$ ，单个凸多边形的点数小于等于 34。图形互不相交，且 Blinker 的出发点和目的地不在图形的边界。