

第五届“图灵杯”NEUQ-ACM程序设计竞赛（个人赛）暨校“蓝桥杯”选拔赛 题目册

比赛开始前不得外泄 2017.12.3

若无说明，题目内存限制均为128M，题目时间限制均为1s。

Coding for fun!

分数分布（计算班级得分时）

题号	分数	题号	分数
A	5	G	30
B	5	H	30
C	10	I	30
D	10	J	50
E	20	K	50
F	20	L	50

简单题 I

A 我爱数学

题目描述

viewsetting特别喜欢数学，但是他算除法的时候特别不熟练，常常会出错。为了维持他对数学的兴趣，你能不能帮他做一下除法运算呢？

输入

第一行是一个整数 T （ $T \leq 1000$ ），表明数据组数。每组数据包括一行的两个整数 a 、 b ，用空格隔开。输入保证在int的范围内，且保证运算合法。

输出

每组数据输出一行 $a \div b$ 的值。保留6位小数。

样例输入

```
2
114 514
1919 810
```

样例输出

```
0.221790
2.369136
```

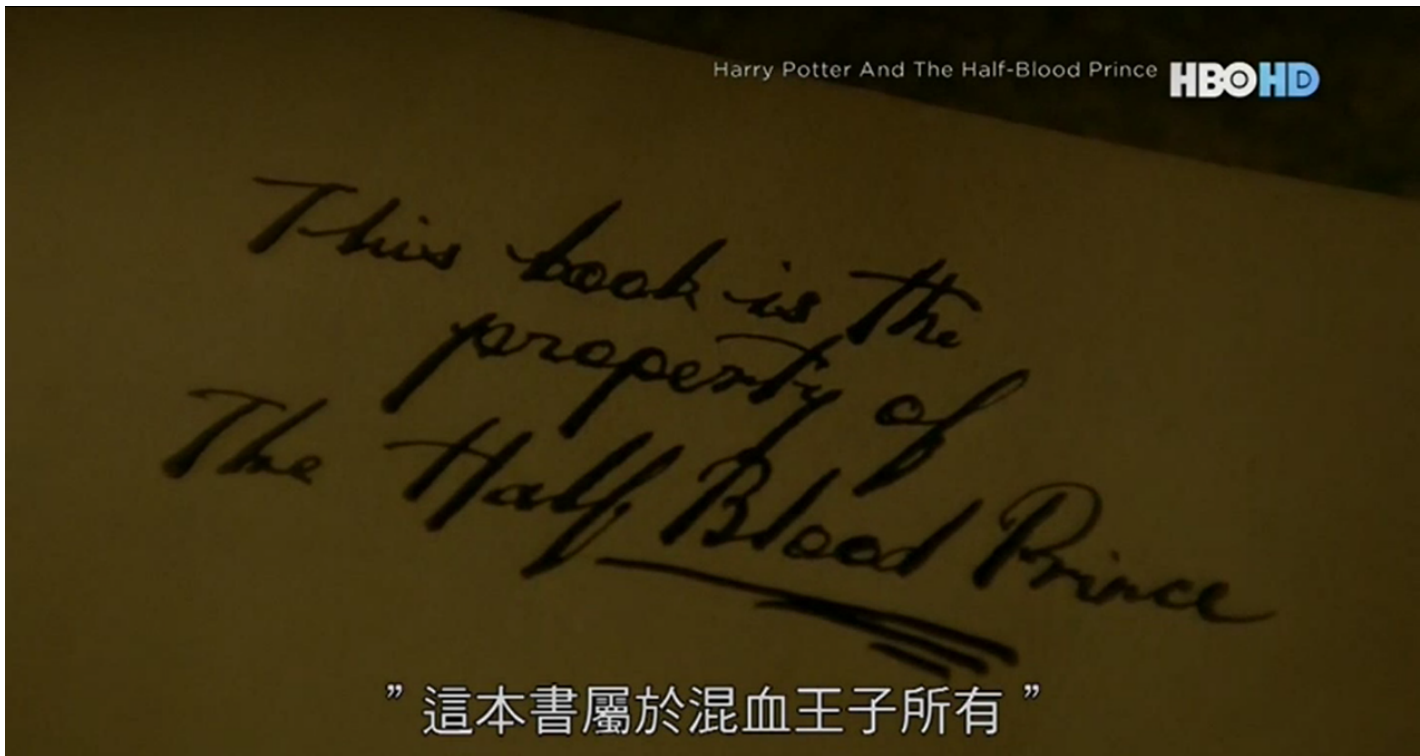
提示

viewsetting会感谢你的！

B 哈利波特与混血王子

内存限制：256M

题目描述



这几天HBO一直在播放哈利波特系列电影，viewsetting觉得还行，就刷了一遍。

哈利波特系列电影改编自J.K.Rowling的同名系列小说。在Harry Potter And The Half-Blood Prince中，哈利·波特和阿不思·邓布利多成功说服了霍拉斯·斯拉格霍恩重返学校，担任魔药课教师，而西弗勒斯·斯内普教授则如愿以偿，成为了黑魔法防御课老师。在第一节哈利·波特还没发到书的魔药课上，霍拉斯·斯拉格霍恩借给了哈利一本旧的魔药课课本。

哈利发现，这本旧课本上面签着“Half Blood Prince”这个名字，此外这本书上还有很多笔记。可是问题来了，这个Half Blood Prince可不是等闲之辈，作为当年的学霸，他的学霸笔记可以让哈利波特的魔药学成绩提高到全班最高，然而他记笔记的习惯特别恶心。

为了防止有人抄他的笔记，Half Blood Prince研制了一套密码笔记书写机制。他使用了一个函数转换他的笔记：对于a-z和A-Z这些拉丁字母，如果令a编号为0，b为1依次类推，z为25。同样地，A为0，B为1.....Z为25。其实Half Blood Prince使用的是 $y = (kx + b) \% 26$ （%是取模运算符）的加密模式，也就是y是笔记上字符编号x对应的真实笔记的字符编号。

那么举个栗子：假如这个密钥的加密模式是 $y = (x + 1) \% 26$ ，那么笔记中的“Bqtrg vhsq akcd.”应该表达的是“Crush with blade.”（注意除了大小写的字母外其他字符不用处理）。那么问题来了，为了让剧情发展下去，哈利波特告诉你Half Blood Prince笔记的加密模式和参数k和b的值，你能设计一个解码程序，帮他还原学霸笔记，击败赫敏登上全班第一的宝座吗？

输入

第一行两个整数k和b,均在一百以内，其中 $k > 0, b \geq 0$ 。

第二行一个长度为l的字符串（ $l \leq 1000$ ），代表Half Blood Prince的一条原始笔记。

注意是多组输入。

输出

这条笔记的真实版本，也就是根据转码规则得到的解码版本。

样例输入

```
1 1
Bqtrg vhsq akzcd.
```

样例输出

```
Crush with blade.
```

C 哈利波特与阿兹卡班的囚徒

题目描述



月亮脸 虫尾巴 大脚板 and 尖头叉子先生
"Messrs. Moony, Wormtail, Padfoot and Prongs..."

Viewsetting又看了一部哈利波特系列电影：Harry Potter and the Prisoner of Azkaban（哈利·波特与阿兹卡班的囚徒）。电影里出现了一个魔法道具：活点地图。

乍看之下，这张地图只是一张空白的羊皮纸；但是当使用者说出“我庄严宣誓我没干好事”（“我庄严宣誓我不怀好意” - I solemnly swear that I am up to no good），墨线就会开始延伸，浮现出一张霍格沃茨的地图（包括秘密通道）。地图上有霍格沃茨里每个人所在的位置，并指示打开秘密通道的方法。使用完毕之后，只要说“恶作剧完毕”（Mischief managed），就可以将地图复原成空白的状态，以防止其他不知道地图口令的人使用。

活点地图是莱姆斯·卢平、小矮星彼得、小天狼星布莱克以及詹姆·波特（哈利的爸爸）共同发明的，专门用来达成恶作剧。他们在就寝时间后，常常化身为动物四处探险，使得他们对霍格沃茨的地下通道有很多了解（布莱克，小矮星和波特是阿尼马格斯，而卢平是狼人。）他们将所发现的秘密通道，制作成活点地图。地图上写著这几位创造者的名字，他们采用仅限彼此所知的昵称（代表他们幻化成的动物）：月亮脸（指身为狼人的卢平），虫尾巴（指小矮星，化身成老鼠），大脚板（布莱克，化身成狗），以及尖头叉子（波特，化身为雄鹿）。而这也是活点地图标题的由来。

哈利波特用这张地图可以说是开了外挂，游戏体验++，然而斯内普发现了哈利有这张地图，并且试图找出其中的秘密，但却被地图羞辱了一顿。卢平教授表示他会调查，就将地图带走了。可是现在哈利波特特别需要这张地图，于是他找到了你，帮他完成这个魔法地图。但是你只是一个麻瓜（不会魔法的普通人类），那里会这些骚操作？但是你可可是会计算机的大手子啊！你就拥有那些魔法师眼中的“黑魔法”！请你编写一个程序，同样也能实现这个地图的功能。

输入

输入一个 n ($n \leq 999$)，代表记录中有 n 个魔法师。接下来 n 行，每行依次输入魔法师的名字，起始坐标和编号。其中坐标 (x, y) 意义是 x 坐标代表东西方向， x 正方向为东。类似的， y 是北正南负且坐标都是整数有序对。编号是三位的，如有前导零则要补足，编号不超过999。

接下来输入一个正整数 k 。然后接下来 k 行对应这地图的记录，如果这个魔法师移动了，依次输入其编号，方向（N为北，S为南，W为西，E为东），步数（正整数）。如果这个魔法师没有移动，格式是编号和“stay”。

注意是多组输入。

输出

和输入格式一样，按照序号从小到大的顺序换行输出魔法师的名字，当前坐标和编号。

样例输入

```
5
Harry (1,1) 231
Hermione (2,2) 402
Ron (1,0) 311
Dumbledore (10,10) 000
Snape (-1,-1) 007
3
231 E 1
402 S 1
007 stay
```

样例输出

```
Dumbledore (10,10) 000
Snape (-1,-1) 007
Harry (2,1) 231
Ron (1,0) 311
Hermione (2,1) 402
```

简单题 II

D Bio Card

题目描述

ingress是一款很好玩的虚 (ji) 拟 (ka) 现 (ceng) 实 (fan) 游戏，每个玩家都会有Bio Card用来和别的玩家线下交换。

抵抗军特工@PJ10086很喜欢这样的收集，每次线下交换完后他都会得到一个卡堆 $\{b_i\}$ 。但是他很懒，只会让他的机器人JunJun来整理到他的收集 $\{a_i\}$ 里。机器人有三种操作模式：

- 1. 切换当前操作的卡片到 a_i
- 2. 在 a_i 后放置目前的卡片 b_{now}
- 3. 抛掉 a_i 之前的卡 a_{i-1} （数据保证 a_i 前面一定有卡片可以抛掉）

一天，队友@lszr不小心把他的机器人弄坏了。@PJ10086很生气，让他整理好自己卡堆。@lszr拜托你写一个程序，整理好他的卡堆。

一开始@PJ10086 会拥有一张自己的卡，即“PJ10086”，并且一开始机器人操作的卡片就是“PJ10086”。

输入

多组输入，以文件结束符（EOF）结束。

每组数据第一行输入一个数m（ $\sum m_i \leq 300000$ ），表示@PJ10086一共收到的卡牌数量。这些卡片的名称是各不相同的。

接下来m行，每行输入一个整数表示操作模式。

当操作数为1或2时，再输入一个字符串s，表明切换到（增加）了哪一张卡片。

操作1的总和不会超过 $\sum m_i$ 的1%。

输出

每组输出一行，表示最终@PJ10086的卡堆顺序，每张卡以空格隔开。

样例输入

```
6
2 lszr
2 andyhui68666
2 KanModel
2 Mouost
1 KanModel
3
3
2 luhongbin
1 luhongbin
3
```

样例输出

PJ10086 KanModel andyhui68666 lszt
luhongbin

提示

该题的输入输出量有可能会很大，请谨慎使用C++的输入输出流。

E 高清LED屏幕

题目描述

补锅猛男设计了一个高清LED屏幕，一切都搞定了，就差一个程序员了。

他麻烦你根据输入的表达式，计算结果。每两个数字或符号之间会有一排"."将其隔开，保证所有计算都在int范围内，且数据均合法。

为了确保屏幕和数据的链接良好，他更是附送了全部数字及符号的图像，方便你进行配置：

[illegible]

输入

两个数字及一个符号，多组输入。

输出

计算结果，同样以字符图像的形式输出。

样例输入

[illegible]

样例输出

.###.
 . . . #
 . . . #
 .###.
 . . . #
 . . . #
 .###.
 ###. . . ###.
 #. #
 #. #
 .###. .###. .###.
 . . . #. #. .
 #. #.
 ###.

提示

该题的输入输出量有可能会很大，请谨慎使用C++的输入输出流。（也许可以呢？）

F PJ的情书

题目描述

寂寞的PJ终于找到了一个超级可爱的女朋友，他想给她写一封情书，但是他文笔不好，情商又低，只能写写代码才能维持的了生活这个样子。

正好PJ可爱的女朋友也是学计算机的，她刚刚学完《计算机导论》，于是PJ灵机一动，打算用二进制写一封情书给她。

因为PJ凑不出来字数，于是PJ想到了这么一种编码方式来**延长**自己的情书长度：

- 1. 先用英语写好情书，记录出现的英文字符的种类 n ($0 < n \leq 52$)，区分大小写。
- 2. 记录每种英文字符出现的频数，按从大到小的次序排好放入队列。如果频数相同，则按字典序。
- 3. 将当前队列中的队首两元素分别作为左右子节点（频数较高的作为左子节点）创建一个父节点，父节点自身的频数值为左右节点的和。并将该父节点放回到队首。
- 4. 重复上述行为直到队列中所有的元素都已经添加到同一棵二叉树上。
- 5. 从根节点开始以深度优先的方式遍历节点，每个节点的左树枝边值记为0，右树枝边值记为1，直到叶节点，按顺序读取树枝的值便是我们所需要的该字符的编码，我们称之为PJ编码。

值得注意的是，PJ希望他的情书可读性更强，所以他**只转换英文字符**，其他字符都不变。并且在输出情书正文之前他要先给他可爱的女孩一个字典以便读懂这篇情书。

输入

一组输入，以文件结束符（EOF）为结束。

输入一整篇英文情书，**包括空格和回车**。整篇情书的长度不超过 10^6 个字符。

输出

前 n 行，以从出现频数高到低的顺序，输出已出现的英文字符和它对应的PJ编码，以英文冒号和空格隔开。

接下来按原来的格式输出已经转换成PJ编码形式的情书。

具体输出格式见样例。

样例输入

```
I AM PJ
I Love you so much!
CC
```

样例输出

```
o: 0000000000000000
C: 0000000000000001
I: 0000000000000001
u: 0000000000000001
A: 0000000000000001
J: 0000000000000001
L: 0000000000000001
M: 0000000000000001
P: 0000000000000001
c: 0000000000000001
e: 0000000000000001
h: 0000000000000001
m: 0000000000000001
s: 0000000000000001
v: 0000000000000001
y: 1
0000000000000001 0000000000010000000001 00000001000000000001
0000000000000001 0000000001000000000000000001000001 100000000000000000000000000001 00100000000000000000 000100000000000001000001!
00000000000000010000000000000001
```

提示

有一位猛男叫做Haffman，他提出了一个结构叫Haffman Tree。

中等题

G cc撩妹

题目描述

大家都知道cc是个英俊帅气的暖男，他最喜欢关心各路好看的妹子了。

这一天，他来到工学馆的一间教室，发现好多好多的漂亮小姐姐。但是小姐姐们都被一些丑陋的男人围着，cc很害怕，他不想得罪他们。

所以cc决定从一个小姐姐开始，把 **所有和她相邻的小姐姐，以及相邻的相邻的小姐姐（相邻小姐姐是指横、竖方向的第一位小姐姐）** 都关心一遍，而不跨过那些丑陋的男人；同时，对他这一次关心到的每个小姐姐，他都会这样操作，直到关心完他能关心到的所有小姐姐为止。

由于不想得罪那些丑陋的男人们，他只能从一个小姐姐开始关心。也就是说，cc没法关心完一片小姐姐后再去关心另一片的小姐姐们。请问cc最多能关心到多少个小姐姐？

输入

多组输入。

每组第一行输入两个数m和n（ $0 < m, n \leq 1000$ ），表示教室的大小，有m行n列的座位。当m和n为0时停止程序。

输入一个教室的人员分布图，用一个矩阵表示：“*”表示小姐姐，“#”表示丑陋的男人。

输出

输出一行一个数字，表示cc最多能撩到的小姐姐数量。

样例输入

```
5 5
####@
#@@#@
#@###@
@@@#@
@@@#@
@@@#@
0 0
```

样例输出

```
8
```

提示

左下方有8个小姐姐cc可以一个一个地关心。

右边一列有5个小姐姐cc可以一个一个地关心。

但是8个小姐姐和5个小姐姐之间都隔着丑陋的男人，所以cc只能选择左下方的8个小姐姐以使得他能关心最多的小姐姐。

H Lethe的手环

题目描述

Lethe学长最近新买了一个手环想送给女友，但他觉得这样太单调了，于是他决定镶嵌一些钻石到手环上。

这个手环有n个镶嵌孔，且这n个孔均匀分布在手环上。因为Lethe有强迫症，所以他买了价值为1、2。。。n的钻石各一个。然而他的女友十分讨厌7这个数字，所以他决定**任意相邻3个钻石的价值之和不能是7的倍数**。现在他不知道有多少种方案，使得每种方案镶嵌钻石的相对顺序不同。你能帮帮他吗？

输入

多组输入，每行一个数字n（ $1 \leq n \leq 12$ ），与题目描述相同

输出

对于每组数据，输出方案数。

样例输入

1
2

样例输出

1
1

提示

注意，**手环是环形的**。所以当n=2时，1 2 和 2 1 是同一种方案，所以答案为1。

I 逃出生天

题目描述：
gold学长从昏迷中醒来以后发现自己被困在一个山洞里，他找了很久，终于找到一个门。门上写着：想要逃出去，只有一个办法 你可以选择一个数n，设m=1 * 2 * ... * (n-1)。如果m是n的倍数，那么门就会自动打开，否则你就别想出去了。 gold学长内心充满了绝望，他想了一些数，但他不知道这些数能不能保证自己逃出去。你能帮助gold学长逃出生天吗？

输入：
第一行一个数T（T<=1000），表示gold学长想的数的个数 接下来每一行一个数n（2<=n<=1e8），表示gold学长想的数

输出：
每行一个输出 对于每一个数，如果学长能逃出去，则输出“escape”（不含引号），否则输出“trapped”（不含引号）

样例输入

4
5
6
7
8

样例输出

trapped
escape
trapped
escape

难题

J Not an easy problem

内存限制：256M

题目描述

谷神有 $n(1 \leq n \leq 100000)$ 个非负整数 a_1, \cdots, a_n 和 m ，满足 $0 \leq a_i, m < 2^{60}, 1 \leq i \leq n$ ，你的任务就是求出满足等式 $\sum_{i=1}^n a_i \oplus k = m$ 的最小非负整数 k 。即

$$\min \left\{ k \in \mathbb{Z}_{\geq 0} \mid \sum_{i=1}^n a_i \oplus k = m \right\}.$$

其中 \oplus 表示异或。如果不存在这样的 k ，则输出 -1 。

输入

多组测试用例。

每组测试用例第一行输入 n 和 m ，第二行输入 n 个非负整数 a_1, \dots, a_n 。

输出

对应每组测试用例输出一行答案，即满足题目要求的非负整数 k ，或者输出 -1 表示不存在这样的数。

样例输入

```
3 5
1 2 3
```

样例输出

```
1
```

K Bang

题目描述

In this problem. You are to simulate a famous wild west-theme card game. Its name is BANG. The game is suitable for four to seven players gather around a table to have a good time. The game is famous partly because it's complicated. Don't worry. You are only to simulate a simplified version of the game.

In our version of the game, only two players are involved. Let's suppose they are Roliygu and Yilan. Each of them has four hit points. A pile of cards is shuffled and placed on the table facing down. The two players draw as many cards from the top of the pile as their hit points(i.e Roliygu draws the top most four cards, and Yilan draws the next four cards). Then the game starts. The game is played in turns, alternatively. Roliygu begins. Each player's turn is divided into three phase:

1. Draw two cards from the top of the pile;
2. Play any number of cards;
3. Discard some cards so that the number of cards in hand does not exceed his hit point.

A card, when played, is discarded. There are some kinds of cards in the game of BANG when played, each of them has its own effect. There are listed as follow:

Bang: You (the player who play card) shoot at your rival. If he failed to avoid it (plays a Miss card immediately), the bullet hits him and his hit point reduce by one. You can use no more than one of this card in one of your turn.

Grenade: Your rival must play a Miss card immediately, or he loses one hit point.

Ghost: Your rival must play a Miss card immediately, or he loses one hit point.

Knife: Your rival loses one hit point immediately.

Miss: You can not play this card in your turn. The only usage of it is to avoid being shot;

Parry: You can not play this card in your turn. But when you have to play a "Miss", you can also play this card to avoid being hurt. Each time you play this card, draw one card from the top of the pile immediately.

When one of the players loses all his hit point, he loses. During the game, each player will follow such strategies:

- In each turn, a player will play as many cards as he can;
- When his rival plays a "Bang"/"Ghost"/"Grenade", he would always play the specific cards if he owns one;
- Play a "Ghost" card prior to all other cards;
- Play a "Parry" card prior to a "Miss" card;
- Discard "Bang" cards first (if he has to), then "Miss" cards, then "Parry" cards;

It is guaranteed that there would be enough cards in the pile to run the game to the end.

输入

Input contains no more than 40 scenarios. Each of them contains multiple lines, each being one card in the pile, from top to bottom. Each scenarios is terminated by a line "===" (three equal mark). The whole input is terminated by a line contains a single period (.).

输出

For each scenario, if the player who makes the first move (i.e Roliygu) wins or loses, output a line "WIN" or "LOSE" correspondingly.

样例输入

Bang
Bang
Bang
Bang
Parry
Bang
Ghost
Knife
Bang
Bang
Grenade
Miss
Grenade
===
.

样例输出

LOSE

L Clow的训练

题目描述



Clow是一个刚上大学的普通计算机专业学生。入学两个月，他喜欢上了同班的Nancy，但一直不知道该如何追求。

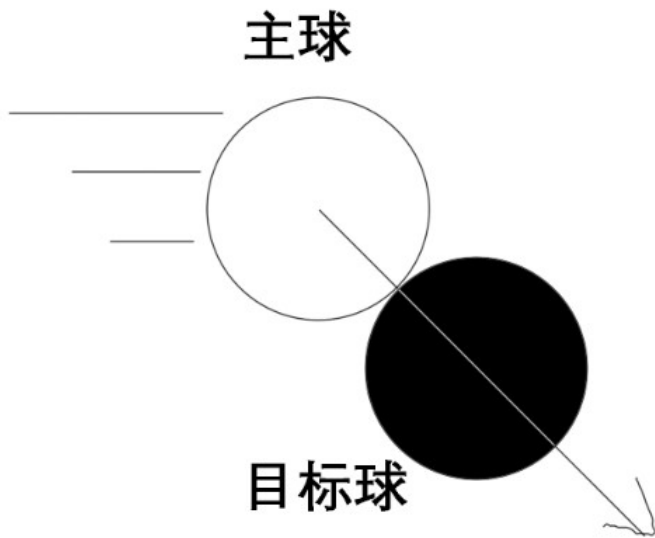
Clow听说Nancy喜欢打桌球，为了能够顺利撩妹，他决定去学习一个。这种游戏是在一张有六个洞的长方形桌子上进行的，桌子四个角各有一个洞，桌子两边长的中间又有一个洞。Clow只是个初学者，不想在乎太多复杂的规则，他现在的目标只是让主球撞击一个他看得顺眼的球让它成功入洞。

Clow问了老板之后得知台球桌是54英寸 x 108英寸，并且这家球馆台球的直径是2英寸。为了方便表示，如图所示，他给台球桌的每个落袋标上了坐标和编号。（见上图）

现在Clow把主球放在一个他舒服的位置上并瞄准了其中一个球，但是他不不确定他能否避开桌上其他球的障碍将他的目标球一杆入洞。你可以帮他解决这个问题吗？

Hint :

球杆可以沿直线方向任意击球，假定两球之间的碰撞是完全弹性碰撞，球被击出后必定沿直线运动。



主球必须直接与目标球相撞，主球不能先撞击其他障碍球或者从桌边反弹再撞击目标球，目标球球心坐标等于球洞中心坐标时被判定为入洞。在该题中主球视为撞完目标球之后就立即从桌上消失了，且不考虑两球相撞后入袋的combination shots或者撞边入袋的bank shots。

详细内容见样例。

输入

多组样例输入。每组测试数据的第一行包含四个实数 X_c, Y_c, X_t, Y_t ：(X_c, Y_c)表示主球位置，(X_t, Y_t)表示目标球位置。

测试数据第二行的第一个整数 N ($0 \leq N \leq 14$)表示桌上其他球（障碍球）的数量，之后是 N 个实数对 $X_1, Y_1 \dots X_n, Y_n$ ，每对(X_i, Y_i)表示障碍球的坐标。

所有 X, Y 满足 $3 \leq X \leq 105, 3 \leq Y \leq 51$ 。不存在重叠的两个球，所有球都保证在桌子内部。

输入以一行仅包含单个数字0的行结束。

所有数据不超过100组。

输出

对每组测试样例，输出一行按照升序排序的数列表示可以被击中的球洞的序号，用空格分隔。

若无法击中任意一个球洞，则输出“no shot”。

样例输入

```
100 10 95 20
0
100 10 95 16
8 5.1 44.6 41.4 10.3 102.9 28.2 99.4 42.2 75.7 32.8 27.9 14.1 82.5 29.5 64 32.5
80 27 40 27
2 38 28 38 26
0
```

样例输出

```
1 4 5 6
1 2 4 6
no shot
```

这是本试题册的结尾。代码改变世界！

Chief Proposition people:CC, Sam X

Proposition Group: CC, Neteal, Nicolas, Jun Pan, Lethe, tiancaiZXP, Turtle, Sam X, viewsetting, Yifan Jing, Zhong He

3rd Version