

BEARTRAP: 小熊和猫陷阱

题目描述

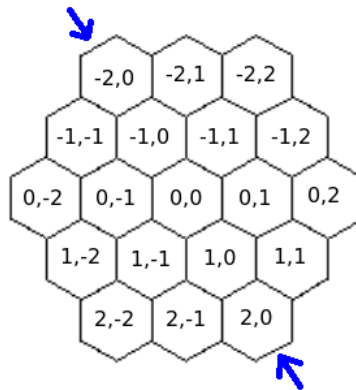
这是一道交互式的题目。每当你输出一行后，需要手动刷新输出缓冲区。在 C++ 中，可以调用 `fflush(stdout)`；在 Java 中，可以调用 `System.out.flush()`；在 Python 中，可以调用 `sys.stdout.flush()`；在 Pascal 中，可以调用 `flush(output)`。只有在你刷新缓冲区后，才能再次从输入中读入。

Limak 是一头小北极熊。他最喜欢的一个游戏是猫陷阱 (*Cat Trap*, 又名 *Black Cat*、*Chat Noir* 或 *Circle the Cat*, 而一个广为人知的中文名称是“围住神经猫”)。游戏的目标是防止地图上的一只猫逃跑。Limak 非常喜欢这个游戏，因为不知道为什么，猫总是会怕它，不会和它玩。你可以自己玩一玩这个游戏：<http://www.gamedesign.jp/flash/chatnoir/chatnoir.html>。

本题中，你的任务是在一个简化版的游戏获胜。简化版的区别在于：

- 游戏地图是边长为 $N = 20$ 的正六边形；
- 初始时没有格子是被堵住的；
- 你必须在 M 个回合中获胜（即，最多可以堵住 M 个格子）；
- 猫按照特定的策略行动，详见下文。

行按照从上到下的顺序标号为 $-(N-1) \sim N-1$ ，最中间的一行编号为 0。列按照从左到右的顺序编号为 $-(N-1) \sim N-1$ ，编号为 0 的列为包含了第一行最左侧格子与最后一行最右侧格子的一列。下图为 $N = 3$ 的地图（蓝色箭头指向第 0 列）：



记 (r, c) 为第 r 行第 c 列的格子。按照上面的定义，与 (r, c) 相邻的格子最多有六个： $(r, c+1)$ 、 $(r-1, c+1)$ 、 $(r-1, c)$ 、 $(r, c-1)$ 、 $(r+1, c-1)$ 和 $(r+1, c)$ 。

边界格子是相邻格子数量小于 6 的格子，共有 $6(N-1)$ 个边界格子。对于任意没有被堵住的格子 X ，我们定义其**边界距离**为从该格子到任意边界格子的最短距离。特别地，没有被堵住的边界格子边界距离为 0，而被堵住的格子以及无法到达边界的格子的边界距离为无穷大。

猫初始时在 $(0, 0)$ 处，即处于地图的中心。游戏的每一回合中：

- 猫进行移动。如果猫处于边界格子中，猫会立刻逃走，你则输掉了游戏。如果猫无法到达任意边界格子（即猫被困在了由被堵住的格子围成的连通块内），那么你赢得了游戏。否则，猫会移动到边界距离最短的一个相邻格子中。举个例子，如果有三个相邻格子被堵住了，两个相邻格子的边界距离为 15，一个相邻格子的边界距离为 17，那么猫会移动到那两个边界距离为 15 的格子中的一个。在游戏一开始时，猫可能向任意相邻的格子移动，因为所有相邻格子的边界距离相等。

- 然后轮到你了。你可以堵住一个格子。你无法堵住一个已经被堵上了的格子，也无法堵住猫所在的格子。

交互格式

输入的第一行包含两个整数 T 和 M ，分别代表测试数据的组数，以及最多可以进行的回合数（回合数对于每组数据单独计算）。

当猫移动时，你需要从输入中读入一行。这一行可能是“WIN”，或者是“CAT r c ”（均不含引号）。如果读到“WIN”，则代表猫已经无法到达边界，你赢了本局游戏，需要开始处理下一组测试数据了（此时你需要从输入中读取新的一局游戏中猫的第一步移动）；当然，如果当前数据是文件中的最后一组，则你的程序应当退出。如果读到“CAT r c ”，则代表猫移动到了格子 (r, c) 。

在猫移动之后，你应当输出一行“BLOCK r c ”，代表你要堵上格子 (r, c) 。请记得输出换行符并刷新输出缓冲区。你无法跳过一回合，也无法堵住已堵上的格子，亦无法堵住猫所在的格子。

数据范围与子任务

子任务 1 (35 分):

- $1 \leq T \leq 10$
- $M = 500$

子任务 2 (65 分):

- $1 \leq T \leq 100$
- $M = 20$

样例数据

输入	输出
2 500	
CAT 1 -1	BLOCK -1 0
CAT 1 0	BLOCK -1 1
CAT 0 0	BLOCK 1 -1
CAT 0 1	BLOCK 0 -1
CAT 1 0	BLOCK 1 1
CAT 0 0	BLOCK 0 2
CAT 0 1	BLOCK -1 2
CAT 0 0	BLOCK 1 0
WIN	
CAT 1 0	BLOCK 5 -12
...	...
...	
WIN	

可视化程序

本题提供了可视化程序，可以单击[此链接](#)下载。你需要使用 Python 3 运行该代码，具体用法请见代码底部的（英文）注释。

样例解释

样例中有 $T = 2$ 组数据，你最多进行 $M = 500$ 个回合。请注意，样例实际上是不合法的，因为猫有时没有移动到边界距离最短的相邻格子中。这一情况不会出现在实际测试中。而第二组数据只是为了展示存在多组数据时的数据格式。

你可以使用可视化程序模拟第一组数据。下图是最终局面的示意图，为了方便展示，假设地图大小为 $N = 4$ (实际数据中 $N = 20$):

```

. / . /
 / . / . /
. , # # # ,
, . # c , # ,
, . # # # .
 / . / . /
 / . / .

```

时间限制

4 秒

Problem Setter: Kamil Dębowski
Problem Tester: Sergey Kulik & Kamil Dębowski
Translated by: Hu Zecong