

题目 0001 – tix (马戏团入场券)

题目来源: USACO Open 2002

推荐者: 刘才良

一、英文原文

Circus Tickets

Author: Gadnell & Backman/Kolstad (1997)

Farmer John is taking 16 cows to see the three ring circus. Being cows, they are seated in the cow section, a 4x4 set of seats near the front of the audience. The seats, rows, and columns are numbered thusly:

```
col 1    col 3
  | col 2 | col 4
  |  |  |  |
  1   2   3   4  <-- row 1
  5   6   7   8  <-- row 2
  9  10  11  12  <-- row 3
 13  14  15  16  <-- row 4
```

The cows are herded helter-skelter into the seats. It is only after they are seated that they check their tickets. In a revelation that will surprise no one, they are not necessarily in the proper seats. The layout of the cow seating is such that they can rearrange themselves only by rotating a row right or left or rotating a column up or down. The examples below show all four operations applied to the first row or column of the proper seating shown above:

4	1	2	3	2	3	4	1	5	2	3	4	13	2	3	4
5	6	7	8	5	6	7	8	9	6	7	8	1	6	7	8
9	10	11	12	9	10	11	12	13	10	11	12	5	10	11	12
13	14	15	16	13	14	15	16	1	14	15	16	9	14	15	16
Rot row 1 right				Rot row 1 left				Rot col 1 up				Rot col 1 down			

Given the initial seating arrangement of the cows, determine a very good set of rotations to rearrange cows so that their tickets and seat numbers match. At least one solution always exists.

Your score for each test case will depend on how close you get to the best number of rotations submitted by other contestants for that case. Your output sequence must be shorter than 500 operations in order to receive credit for this problem.

HINT: The sequence 1l 1l 1l 4u 1r 4d 1l 1l 4u 1r 4u 4u 4u swaps the upper left cow with the cow to her right.

INPUT FORMAT:

Four lines, each with four space separated integers. Line 1 is the first row; line 2 is the second row; and so on.

SAMPLE INPUT (file tix.in):

```
4 1 2 3
6 7 8 5
10 11 12 9
14 15 16 13
```

OUTPUT FORMAT:

A series of output lines that contain the ordered sequence of rotations that will arrange the cows. Each line contains a row or column number (1..4) followed by a space followed by a lower-case letter that indicates the operation to be performed: 'r' for rotate a row right, 'l' for rotate a row left, 'u' for rotate a column up, and 'd' for rotate a column down.

SAMPLE OUTPUT (file tix.out):

```
1 l
2 r
3 r
4 r
```

[Other correct sequences are equally valid though might not garner as many points.]

二、中文翻译

马戏团入场券

Farmer John 要带 16 只母牛去看杂技，母牛们将坐在观众前附近的一个 4x4 座的牛区。所有座位、行、列如图标上数字。

```
1 列    3 列
| 2 列 | 4 列
|  |  |  |
```

```

1  2  3  4  <-- 1 行
5  6  7  8  <-- 2 行
9 10 11 12  <-- 3 行
13 14 15 16 <-- 4 行

```

母牛们随机坐在位置上，等它们坐下之后才进行检票。很明显，它们没有必要坐在合适的位置上，位置已被设计成可以左右转一行或上下转一列。下面的例子说明了由上面的状态经过 4 种转动操作所到达的目标状态。

4	1	2	3	2	3	4	1	5	2	3	4	13	2	3	4
5	6	7	8	5	6	7	8	9	6	7	8	1	6	7	8
9	10	11	12	9	10	11	12	13	10	11	12	5	10	11	12
13	14	15	16	13	14	15	16	1	14	15	16	9	14	15	16
右转第 1 行				左转第 1 行				上转第 1 列				下转第 1 列			

给定牛的原始位置，用较少的旋转次数将牛重排，使得他们的入场券与座位号相符。无论原始位置如何，总有至少一种可行方案。

你的每个测试点的成绩取决于你的方案与所有选手的最优方案的接近程度。你的输出序列必须少于 500 个操作。

提示：序列 `1l 1l 1l 4u 1r 4d 1l 1l 4u 1r 4u 4u 4u` 交换了左上角的牛与它右边的牛。

输入格式：

4 行，每行 4 个数字。第一行是坐在第一行的牛，第二行是坐在第二行的牛等等。

输出格式：

输出按操作序列排序。每行一个行号或列号(1..4)，接下来是一个空格和一个小写字母。
'r'代表右转，'l'代表左转，'u'代表上转，'d'代表下转。

输入样例(tix.in)：

```

4 1 2 3
6 7 8 5
10 11 12 9
14 15 16 13

```

输出样例(tix.out)：

```

1 l
2 r
3 r
4 r

```

[其他正确的序列同样有效，但可能不会得太多的分。]

三、初步讨论情况

邵烜程：

可以先按照提示内容通过不断交换两个母牛，得到一个可行序列，然后简化。规则可以有：
 $1l\ 1l\ 1l \rightarrow 1r$ ； $1r\ 1l \rightarrow$ 抵消；etc.

方奇：

只想到搜索。如果只是求次优解，可以在广度搜索中加入一些估价函数（类似八数码问题，用当前各数字与目标位置距离构造函数），这样能加快出解速度并求得次优解。

高正宇：

我本来是构造法+随机做的，结果几乎没有什么分。要得分高，可能只好 A*了。可以做做看

雷环中：

1. 归纳思考许多个两个位置的组的近乎最优的转换序列，然后组合起来----基本变换
2. 先用四种基本操作完成 1, 2, 3 行的归位，最后对第四行采用基本变换，即（1）得出的结论。
3. 对结果序列进行等价缩短变换。
4. 将目标状态稍微变换一点，与（2）配合，这样常常会得到更优的解。

此题状态很多，似乎不像 DP。搜索可以较好的解决这道题，但组织方式和优化方向很多。很好的一道题，开放性较大，不过情况很多，实现起来麻烦。编出一个一般的程序是不难的，但要想编出一个相当接近最优解的程序我认为很难。此外，单是对结果序列进行等价缩短变换就是一个不错的问题，贪心算法常常意义不大，有进一步探讨的余地。很值得讨论。

详细说明：

较容易想到的算法是归纳思考许多个两个位置的组的近乎最优的转换序列（如水平相邻的两个，对角线相邻的两个），然后组合起来，但这样的总步数是相当多的。于是较好的方法是先用四种基本操作完成 1, 2, 3 行的归位，最后再对第四行采用如前所述的基本变换的方法。最后，还可以对结果序列榨点汁出来——相邻几个转换操作的等价缩短变换，但这样的效果相对而言不是很大。此外，我们也可以将目标状态稍微变换一点（枚举末状态），看是否可以得出更短的转换序列，因为问题的规模 4×4 很小，可用时间来换取对最优解的尽量逼近。同理，也可以枚举初状态，优中选优。

张云亮：

较繁琐，没有什么好算法，先宽搜 6,7 步，实在不行只好一格格的移动了。

刘才良：

可以说较优解是比较多的，于是可以用广度搜索，并在搜索过程中放弃一些被认为比较差的节点，使总结点数呈线性增长。这种折衷的方法可以保证很快出解，并且比较接近最优解。

姜尚仆：

如果根据提示用最简单的构造，结果可能会过大；也可以用搜索；或是综合一下，大体上用搜索，局部用构造。

刘汝佳：

刚才几位集训队员虽然大都说得比较笼统，但是是有启发性的。我们可以把思路总结如下：

- 基本框架：随机化产生，启发式搜索，双向搜索，构造。
- 可选策略：结果调整（收缩），过度剪枝（保留价值大的节点）

我们可以把这些方法综合起来，例如先搜索，后构造，最后调整结果。搜索可以是启发式的，还可以是双向的，节点生成可以是随机的，节点保留也可以是启发式的，让节点数不过大...

希望各位同学继续分析，并希望有同学能认真的进行研究，例如可以比较各种方法的优劣并提出新的方法。