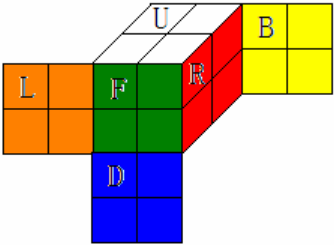


算法实现题 5-21 2×2×2 魔方问题

★问题描述:

2×2×2 魔方的构造如图所示。图中英文字母 U, L, F, R, B, D 分别表示魔方的 6 个面中的上面, 左面, 前面, 右面, 后面, 底面。魔方的每个面都可以绕其中轴旋转。给定魔方的初始状态, 可以经过若干次旋转将魔方变换成每个面都只有一种颜色的状态。绕中轴将一个面旋转 90 度算作一次旋转。试设计一个算法计算出从初始状态到目标状态 (每个面都只有一种颜色的状态) 所需的最少旋转次数。

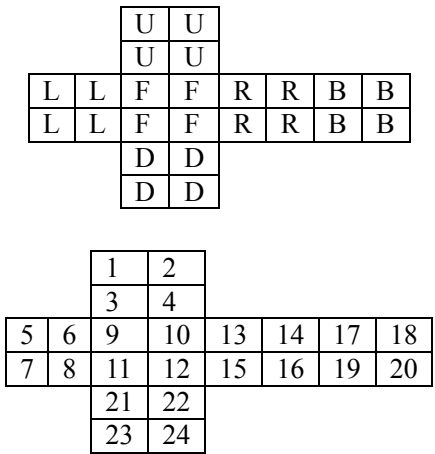


★编程任务:

设计一个算法, 对于给定的 2×2×2 魔方的初始状态, 计算从初始状态到目标状态 (每个面都只有一种颜色的状态) 所需的最少旋转次数。

★数据输入:

由文件 input.txt 给出输入数据。2×2×2 魔方目标状态的每个面都有一种颜色, 分别用大写英文字母 W, O, G, R, Y, B 来表示。将 6 个面展开并编号如下图所示。文件将给定魔方的初始状态的 6 个面, 按照其编号依次排列, 共有 12 行, 每行有 2 个表示方块颜色的大写英文字母。



### ★结果输出:

将计算出的最少旋转次数和最优旋转序列输出到文件 `output.txt`。文件的第一行是最少旋转次数；接下来是最优旋转序列。用表示各面大写英文字母表示每个面的顺时针旋转；表示各面大写英文字母紧接一个“-”表示每个面的逆时针旋转。如果不存在满足要求的旋转序列则输出“No Solution!”。

#### 输入文件示例

```
input.txt
G W
G R
W W
O O
O G
B G
B R
B R
Y R
Y W
Y O
Y B
```

#### 输出文件示例

```
output.txt
2
R U
```