

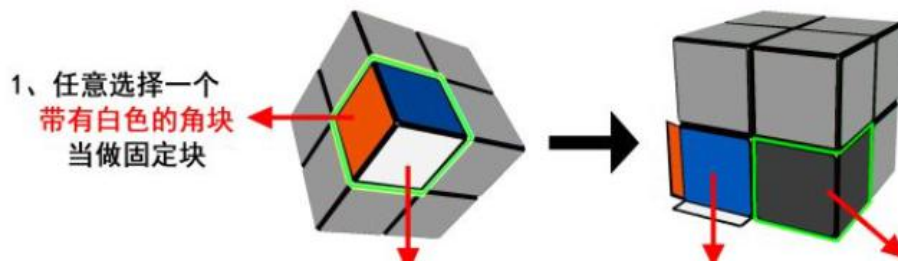
二阶魔方（Pocket Cube）算法

1 还原算法 solve

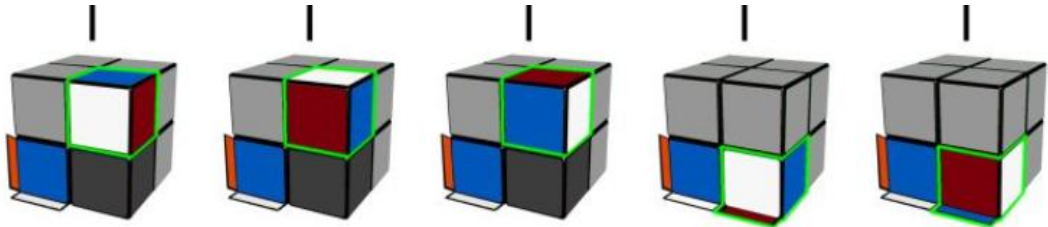
参考教程：<https://zhuanlan.zhihu.com/p/61016023>。

1.1 第一步 还原下面（即白色面）和底层

拿到一个打乱的二阶魔方后，第一步就是先找到任意一个带有白色的块并将该块放在左下角且要白色向下，将该块命名为固定块，如下图所示。



然后找到该固定块右侧本来应该放的块，看看在什么地方，命名为“待还原块”，然后将其移动到如下所示五种情况之一。



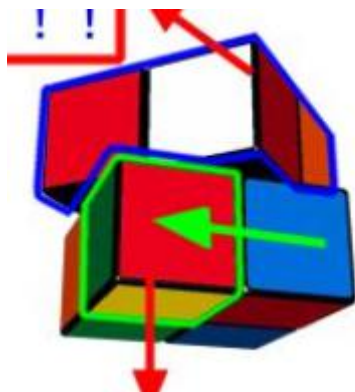
然后使用“左上右下”方法将“待还原块”还原到右下角。
重复上述步骤直到底面都是白色而且前后左右的下层都是一样的。

1.2 第二步 还原上面（即黄色面）

此时将白色面向上放置，然后开始寻找带有黄色的块，将其放在前方右下角，如下图所示。



然后使用“左上右下”公式直至该块复原（即黄色向下）。
然后将底层顺时针旋转，如下所示。



重复以上步骤直至黄色面全部复原。此时黄色向下，白色向上，靠近白色的侧面八个也都复原了，只有靠近黄色的侧面八个没有复原。

1.3 第三步 还原靠近黄色的侧面八个

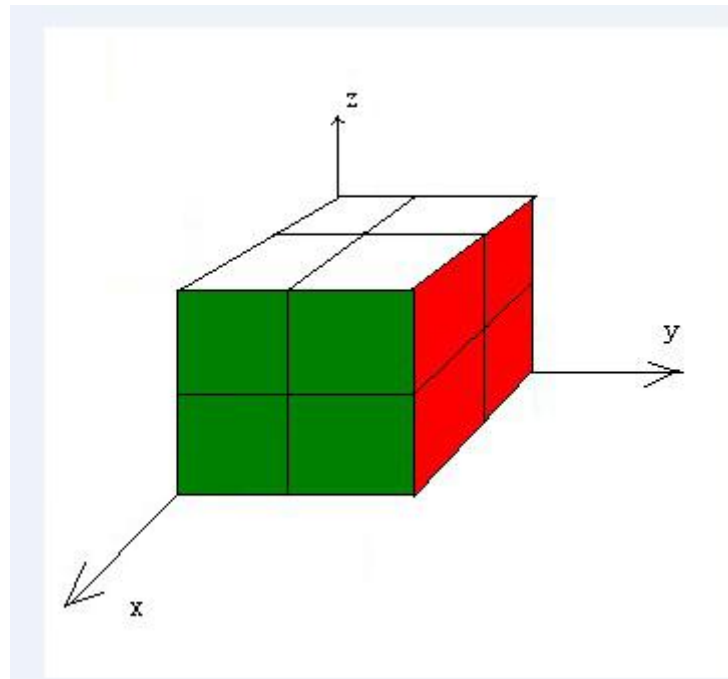
我们一开始规定的是白色为下面，黄色为上面，但是此时我们要把白色向上黄色色向下，因此我就不说上下了，直接说“靠近黄色的侧面八个”。

此时将白色面向上，黄色面向下，开始寻找是否有已经复原的侧面，如果没有就试着执行“左上右下”中的“左”这一步，执行 1-3 次若都没有也没关系，直接执行下一段操作就会出现复原好的侧面。如果有复原的侧面，则将该面朝右，然后执行下一段。

“左上右下”执行三遍，然后把下层（即黄色那一层）顺时针旋转一下，然后再“左上右下”执行三遍，然后再把下层（即黄色那一层）顺时针旋转一下，然后再“左上右下”执行三遍，然后再把下层（即黄色那一层）顺时针旋转两下，然后再“左上右下”执行三遍。

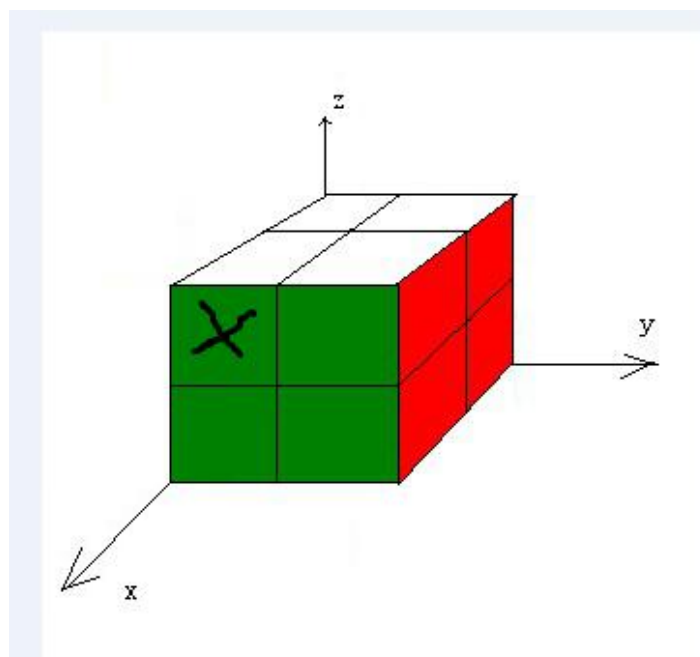
此时稍微调整，即执行“左上右下”中的“左”即可完全复原。

2 数据结构



我们先定义颜色，0-5 分别表示橙蓝红绿黄白六种颜色。四维数组 `cube[2][2][2][6]` 用来存储 8 个方块共 24 面的颜色。其中前三个维度为方块的坐标，分别为 (x, y, z) ，第四个维度为该块三个面的位置，我们对第四个维度规定如下：0 表示朝前的面，1 表示朝右的面，2 表示朝后的面，3 包表示朝左的面，4 表示朝上的面，5 表示朝下的面。

例如，下图中打叉的面可表示为 `cube[1][0][1][0]=4`。



下图中打叉的面可表示为 `cube[1][1][0][1]=2`。

