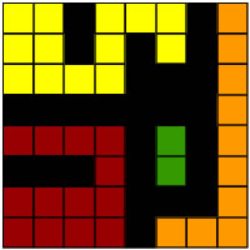
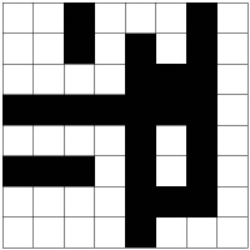
连通域标记算法

原理

对于一个二值图像, 其前景为1, 背景为0, 我们希望给其相连的前景部分标记为同一个label, 不同的连通域标记为不同的label, 我们希望得到的最终结果如图2所示。



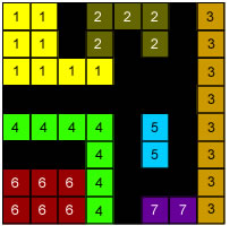
第一阶段

我们从图像的左上角点遍历整个图像。检查每一个像素点的上边和左边有没有已经标记好的label,

1. 如果没有,标记为1。
2. 如果它只有左边有标记，或者左边与上边的标记一样， 那么标记为与左边同样的label,
3. 如果左边与上边的标记不一致, 给该像素标记两个label中较小的那个, 同时让大的label隶属于小的label

数据结构： label的标记可以存在一个二维数组中, 第一行的下标就是label, 第二行表示该label是否具有继承关系, 有则存放父类的下标, 没有则为-1;

遵循上述规则, 我们已经标记完图像上的每一个像素, 并给他们赋了一个值



唯一的问题在于有些连通域我们看着是一个, 但是却给了它两个label, 但是好在我们知道他们label的继承关系, 因此第二遍就是把具有继承关系的部分给合并成一个。

第二阶段

将上一步得到的二维数组的第一行从左到右遍历, 如果该label的第二行不为-1,即存在继承关系, 则将该位置的第一行改为父label。如果第二行为-1则保持不变，遍历完成后， 再从左上角点开始遍历图像，只需按像素点位置的label查找并将其值改为二维数组相应位置的label。(可以在遍历的时候将每个相同label的坐标点放入相同的容器, 可用于连通域计算)

二维数组中label的不同数值的个数就是连通域的个数。

