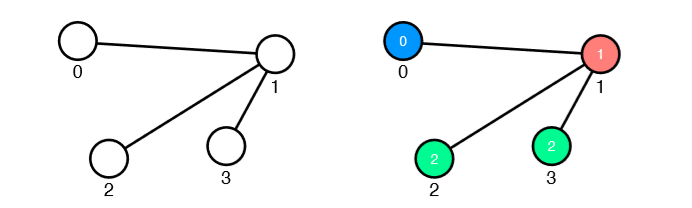
问题一： 图的着色

设计一种算法，找出图的最少着色方法。 鉴于提供的图，给图中的每个结点上色，着色限制是任意相连的结点的颜色不同­­­­，求使用最少颜色的方法。如下图：一个包含4个节点的图，及其对应的一种着色法，实际上，3种颜色并不是最优的方法，最优的方法是2种颜色



图表示节点集合， 表示边的集合

输入：行信息， 第一行表示 结点数目 和 边的数目 ， 从第二行开始，每行表示一条边的信息

输出：第一行表示最少的颜色数量 ，第二行，每个结点的颜色

例子：

输入：

4 3

0 1

1 2

1 3

输出

3

0 1 2 2

着色方法：

1. 对未上色的点列表U做排序，排序规则：根据每个点的未着色邻接点数目，从大到小
2. 选一个未用的颜色i，开始一轮着色，同时准备一个集合 , 称之为用颜色i上色的点集
3. 对U中的每个点依次开始尝试上色，如果此点不与中的任何点相连，那么给它用i上色，并把它加入，否则跳过此点
4. 把里的所有点从U里移除
5. 重复进行1-4，直到所有点完成着色

着色示例详细讲解（以前面的图为例）：

第一轮：

1. 排序后点列表U是（1,0,2,3）,因为开始所有点都未上色，1号点未着色邻接点数目是3，而0号，2号，3号点的未着色邻接点数目都是1，所以排序后U是（1,0,2,3）;当然，0号，2号，3号点的顺序怎么排都行，没啥区别。
2. 选择一个新颜色，比如0号颜色，准备上色，为空。
3. 依次扫描U, 扫描1号点，发现1号点不与中的任何点相连，给1号点用0号颜色上色，并把1号点加入；然后继续扫描0号点，发现0号点与中的1号点相连，即不能用0号颜色给0号点上色，跳过；同理，继续扫描2号点，3号点，都发现和中的1号点相连，均跳过
4. 把中的1号点从U里移除，U列表变为（0,2,3）

第二轮：

1. 再次排序后列表U是（0,2,3），因为现在0号，2号，3号点的未着色邻接点数目都变为0，继续保留了第一轮的顺序，当然其他图有可能会产生排序变化
2. 选择一个新颜色，比如1号颜色，准备上色，为空。
3. 依次扫描U, 扫描0号点，发现0号点不与中的任何点相连，给0号点用1号颜色上色，并把0号点加入；然后扫描2号点，发现2号点不与中的0号点相连，可以用1号颜色给2号点上色，并把2号点加入；继续扫描3号点，发现3号点不与中的0号，2号相连，也可以用1号颜色给3号点上色，并把3号点加入
4. 把中的0号，2号，3号点都从U里移除，U列表变为空，着色完毕

输出上面算法的最终结果：（上面最终结果为1号点是0号颜色，0号，2号，3号点都是1号颜色）

2

1 0 1 1