**暨南大学本科实验报告**

课程名称 算法分析与设计 成绩评定

实验项目名称 第六周课程实验 指导教师 李军

实验项目编号 01 实验项目类型 设计 实验地点 机房

学生姓名 倪煜麟 学号 2016054314

学院 电气信息学院 专业 软件工程

实验时间 2019年 3月 4 日

**一、实验目的**

* 学习、熟悉回溯算法；
* 学习回溯的思想；

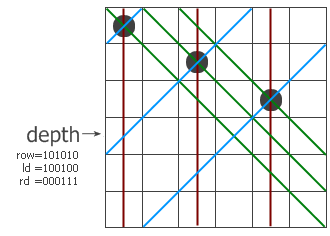
**二、实验环境**

Dev C++ 5.7.1 编程环境。

**三、实验内容**

1. n后问题（5.5节）

对于回溯法解n后问题，在5.5节ppt所涉及的算法基础上，我们可以引入位运算提高算法效率。



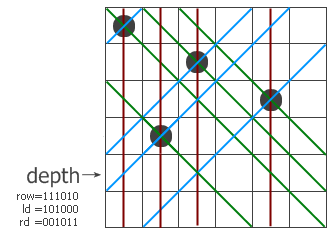
如n=6时，对于已有状态如图，引入3个参数row、ld、rd，分别表示对于depth=4时（即第4行的情况），纵向1、3、4不可放置棋子（第四行红线出现的列），左下-右上对角线（第四行蓝色出现的列）1，4不可放置棋子，左上-右下对角线同理可得。

此时pos = !(row|ld|rd) = !(101111) = 010000。意为第depth=4行，第二个位置可以放置棋子。

状态转移depth+1（下一层递归）

**row** += pos; **ld** = (ld+pos) << 1; **rd** = (rd+pos) >> 1;即可。

效果如图：



当pos = 000000时，说明当前状态不可放置棋子，此时回溯，回归上一个状态。当满足row=111111时，即全部填满，此时便获得可行解。

代码见queen.cpp。

1. 图的m着色问题 （5.6节）

典型的DFS问题，追求效率图可以使用前向链式星或链表存储，本人在此为方便理解和操作，使用邻接矩阵存储。

DFS维护参数：当前结点。

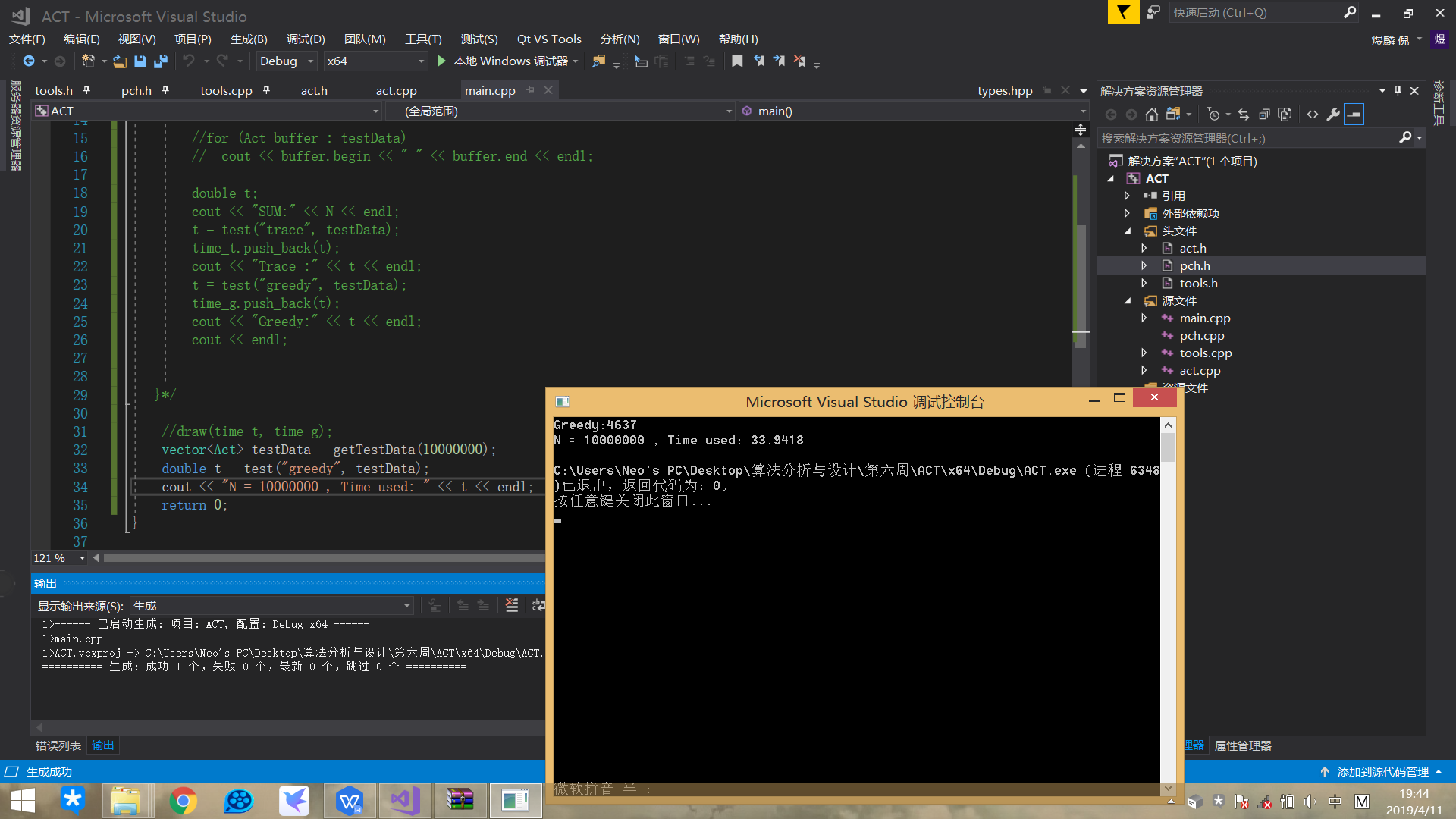
边界条件：对于第k个结点，与其相连的所有节点若已经染色，且颜色相同，则第k个结点不可染当前颜色。

DFS返回条件：N个结点均被染色

总结：DFS样板题，确定维护参数、边界条件、返回条件即可闭着眼睛写。

代码见mColor.cpp

DFS深度N，每一层可能的情况M种，时间复杂度M^N。

1. 活动安排问题（5.8节）
2. 大家把该算法的原理弄清楚，把程序看懂。它的算法复杂度为什么是O(n!)，请你用通俗的语言解释或证明。
3. 因本次实验过于简单，故在此一并附上贪心法对比，本实验使用openCV绘制曲线图，并使用openCV中getTickCount方法作为计时方法，精度达到1\*10-5级，高于一般计时方法。
4. 时间复杂度分析：
5. 回溯法：N个任务，每次搜索都要与N-1个任务组合，之后进入下一层，再与N-2个任务组合，如此反复，直到检测所有任务。所以复杂度为N！。
6. 贪心法：N个任务，对于所有任务排序需要N\*log2N,随后即为线性操作（O(1)），即遍历N个任务一次，故总耗时可认为即是N\*log2N。
7. 编写一个随机生成测试数据的子程序，参数是n,它产生n个活动，存到数组A中，与原算法对接起来。
8. 逐渐增大n值进行测试，直到计算机求解比较困难为止。
9. 回溯法测试数据量在120个时，耗时已经达到18秒之多！贪心法测试数据量在1000W个时，耗时33秒多，100W个时，耗时3秒多。
10. 
11. 显然符合指数级与N\*log2N级的差距。曲线见后附图。
12. 绘图。横轴为n,纵轴为耗时。

这些数据在贪心算法中还要用到。

