**实验二 递归算法求解**

1. **实验目的**

1.掌握非递归算法的设计方法及数学分析框架。

2.掌握递归算法的设计方法及数学分析框架。

3.理解递推和递归的关系，理解哪些问题可以用递归算法来解决

1. **实验要求**
2. 熟悉编程环境；
3. 能把算法的伪代码编成程序。
4. 能得到正确的程序运行结果
5. **主要仪器及耗材**

计算机及相关软件

1. **实验内容**

1. 欧几里得算法是否可以递归求解，如果可以，请设计欧几里得定理的递归算法，写出递推式，并进行求解，最后进行编程实现。

**必然是可以的！gcd(m,n)=gcd(n,m%n) 已实现**

2.以下算法是求一个十进制正整数的二进制位数的算法。

算法 BinRec(n)

//输入：一个正的十进制数n

//输出：n的二进制表示的位数

if n=1 return 1

else return BinRec(└n/2┘)+1

**问题：**

1. .写出次算法的递推式，并进行求解？

**BinRec(n)=BinRec(floor(n/2))+1**

②.实现此算法的非递归和递归两个版本。

**OK**

③. 编写程序，实现以上算法，得出正确的结果。

**ok**

3.考虑下面的递归算法。

算法：Riddle(A[0..n-1])

//输入：包含n个实数的数组A[0..n-1]

if n=1 return A[0]

else temp←Riddle(A[0..n-2])

if temp ≤ A[n-1] return temp

else return A[n-1]

**问题：**

1. 该算法计算的是什么？

**找到n个元素中最小的一个元素。**

②.建立该算法所做的基本操作次数的递推关系并求解。

1. . 编写程序，实现以上算法，得出正确的结果。

**OK**

4.考虑下面的算法来检查一个由邻接矩阵表示的图是否是完全图。

算法: GraphComplete(A[0..n-1,0..n-1])

//输入：一个无向图G的邻接矩阵A[0..n-1,0..n-1]

//输出：如果G是完全图，返回1，否则返回0

if n=1 return 1

else

if not GrapComplete(A[0..n-2,0..n-2]) return 0

else for j←0 to n-2 do

if A[n-1,j]=0 return 0

return 1

**问题：**

1. 这个算法最坏情况下的效率类型是什么？

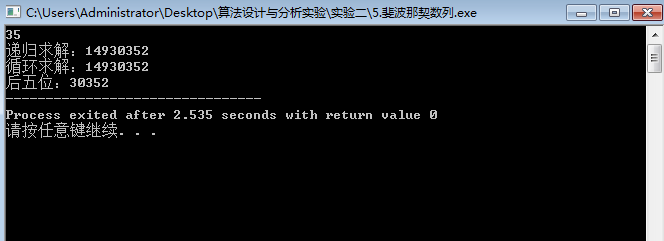
**O（N\*N）**

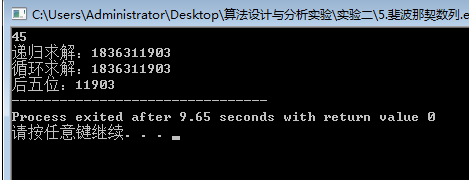
②. 编写程序，实现以上算法，得出正确的结果。

**OK**

1. 选择一种语言，计算第n个斐波那契数列的最后5位数。要求分别实现下列两种算法：（a）基于定义的递归算法F(n)，（b）基于定义的循环算法Fib(n)。做一个实验，看看在你的计算机上，这两个程序一分钟能处理的最大的n是多少？

**已分别实现，大数应该用数组去存比较好，这里简单用了long来测试，这取决于数据类型的最大表示范围。**





1. **注意事项**
2. 我们应该谨慎的使用递归算法，因为它们的简洁可能会掩盖其低效率的本质。
3. 注意设计递归时问题需要满足的条件。
4. 熟悉掌握递归的数学分析方法。