

# 安徽财经大学实验报告

班级：\_\_\_\_\_ 学号：\_\_\_\_\_ 姓名：\_\_\_\_\_ 实验成绩：\_\_\_\_\_

课程名称：acm 程序设计实验 实验名称：综合练习 2

实验地点：信息管理实验室 日期：6-7 时间：19:00~20:30 教师：张林

|       |  |    |   |
|-------|--|----|---|
| 实验目的  | 1. 了解不同算法之间效率比较<br>2. 综合练习不同算法   |    |   |
| 实验环境  | 软件平台：Visual C++ 6<br>操作系统：Windows xp<br>硬件：Core2 2.93G 2M 内存   |    |   |
| 实验原理  | 利用穷举的方法，将 x 在指定的范围内的所有值从小到大列举出来，选择最小的并且满足题意的 x 返回  |    |   |
| 实验内容  |  |    |   |
| 题目    | Problem F  | 编号 | 1.  |
| 题目描述  | <p>勇敢的 Sbarrow 船长驾着白珍珠号，环游大半个地球，历经艰险，终于找到了琼斯的宝箱。可惜宝箱异常坚固，怎么都打不开，上面只有四个每隔一分钟变化一次的整数，还有一个用来输入密码的锁。Sbarrow 尝试了很多密码，但都不对。不过一天夜里，他得到了启迪，卡里布索女神托梦给他，告诉他：盒子的密码是不断变化的，如果设某一时间盒子上的四个整数，从左到右分别为 l,h,a,b 则此时盒子的密码为满足以下三个条件的最小的整数 x：</p> <p>1. <math>l \leq x</math> 且 <math>x \leq h</math></p> <p>2. <math>a \bmod x = 0</math></p> <p>3. <math>x \bmod b = 0</math></p> <p>Sbarrow 虽然勇敢，也很聪明，但他可不是一个优秀的算法专家，现在请你来帮他解决这个问题吧。</p> |    |   |
| 程序源代码 | 详细代码   |    | 关键功能注释                                    |
|       | <pre>#include "stdio.h" #include "stdlib.h"  int main() {     int deal(int,int,int,int);</pre>   |    | //main() 函数读取 input.txt 文件中的测试数据，并将测试结果写到 |

|      |   |   |
|------|---|---|
|      | <pre> int test_number; int i,result; int high,low,a,b; FILE *in,*out;  if ((in = fopen("input.txt","r"))== NULL){     printf("input file open failed!\n");     return -1; } if ((out = fopen("output.txt","w"))== NULL){     printf("output file open failed!\n");     return -1; }  fscanf(in,"%d",&amp;test_number); for (i = 0;i &lt; test_number;i++){     fscanf(in,"%d%d%d%d",&amp;low,&amp;high,&amp;a,&amp;b);     result = deal(low,high,a,b);     fprintf(out,"Case #%d: %d\n",i + 1,result); }  fclose(in); fclose(out); return 0; }  int deal(int low,int high,int a,int b) {     int x;      for (x = low;x &lt;= high;x++){         if (a % x == 0){             if (x % b == 0){                 return x;             }         }     }     return -1; } </pre> | <p>output.txt 文件中</p> <p>//打开 input.txt 和 output.txt 文件,并检查是否成功打开</p> <p>// 读取测试组数 (test_number)</p> <p>//读取 x 的上下限 (low, high) 和能被 x 整除的数 a, 能整除 a 的数 b</p> <p>//将满足条件的 x 写入 output.txt 文件中</p> <p>//关闭已打开的文件</p> <p>//deal() 函数找出符合条件的 x</p> <p>//穷举制定范围内的 x, 找出符合条件的最小值并返回</p> |
| 程序编译 | 错误或警告信息   | 解决方法  |
|      | 无   | 无   |
|      | 无   | 无   |

| 测试<br>结果<br>与分<br>析 | 输入数据   | 输出数据               | 分析   |
|---------------------|--|--------------------|--|
|                     | <b>5 10 18 3</b>   | <b>Case #1: 6</b>  | $5 < 6 < 10 \ \&\& \ 18 \bmod 6 = 0 \ \&\& \ 6 \bmod 3 = 0$          |
|                     | <b>19 21 18 3</b>  | <b>Case #2: -1</b> | 不存在 $19 \leq x \leq 21$ 的 $x$ 使得 $18 \bmod x = 0$ 和 $x \bmod 3$ 同时成立 |
| 题目<br>小结            | 利用穷举法可以“方便的”的找到问题的解决方案   |                    |  |
|                     | //以下可以复制每个题目区域   |                    |  |
| 实验<br>心得<br>和总<br>结 | 1. 在穷举对象数目可以忍受的范围内，穷举不失为一种不错的解决问题的想法<br>2. 对于穷举对象过多的问题可以通过增加筛选条件有效减少穷举对象的数目，以达到较好的效果 |                    |  |