课程设计题目（2024版）

**课程设计报告要求：**

**1.所有的课程设计报告，均要有封面，包括：数据结构课程设计、班级、学号、姓名、和指导教师；**

**2.目录；**

**3.对每一题，给出自己采用的数据结构；**

**4.给出算法设计思想；**

**5.给出实现的源程序，并在必要的代码处给出注释；**

**6.给出测试数据和结果；**

**7.给出算法的时间复杂度、另外可以提出算法的改进方法；**

**8.结束语：说明总体完成情况，每一题的程序代码行，总代码行，心得体会；**

编程语言：C、C++ 等

检查方式:

1. 总体上检查程序的代码量，正确性，可读性，健壮性，功能的完备性,程序的结构是否合理；局部检查三个以上函数块
2. 检查程序的运行情况
3. 检查时间：每个学生的检查时间10分钟左右

时间安排包括：

1 上机时间安排

2 课程设计报告电子文档上交时间

3 课程设计检查时间

**（1-9题必做， 10-20任选1题）**

**1、系统进程统计（必做）（链表）**

[问题描述]

　　设计一个程序，每秒统计一次当前系统的进程状况，并按照内存使用自多到少排序打印输出相关信息。对已经结束的进程，另外给出一个列表，并显示该进程的结束时间和持续时间。

[基本要求]

（1） 该题目要求使用两个链式线性表。一个链表存储当前活动进程，要求使用单向链表，并按照内存使用自多到少排序。另外一个链表存储已结束进程，要求使用双向链表，按照持续时间自少到多排序。

（2） 每秒在窗口内更新一次当前系统进程情况，输出内容包括：进程名，持续时间，内存使用情况。

（3） 每秒在窗口内更新一次已结束进程情况，输出内容包括：进程名，持续时间，结束时间。

（4） 注意进程在这两个链表中的切换，一个进程既可被结束，也可以过一段时间后再被运行。

**2、迷宫问题（必做）（栈与队列）或者八皇后问题**

[问题描述]

利用栈操作实现迷宫问题求解。

[基本要求]

（1）从文件中读取数据，生成模拟迷宫地图，不少于20行20列。

（2）给出任意入口和出口，显示输出迷宫路线。

**3、家谱管理系统（必做）（树）**

[问题描述]

实现具有下列功能的家谱管理系统。

[基本要求]

（1）输入文件以存放最初家谱中各成员的信息，成员的信息中均应包含以下内容：姓名、出生日期、婚否、地址、健在否、死亡日期（若其已死亡），也可附加其它信息、但不是必需的。

（2）实现数据的文件存储和读取。

（3）以图形方式显示家谱。

（4）显示第n 代所有人的信息。

（5）按照姓名查询，输出成员信息（包括其本人、父亲、孩子的信息）。

（6）按照出生日期查询成员名单。

（7）输入两人姓名，确定其关系。

（8）某成员添加孩子。

（9）删除某成员（若其还有后代，则一并删除）。

（10）修改某成员信息。

（11）要求建立至少40个成员的数据，以较为直观的方式显示结果，并提供文稿形式以便检查。

（12）界面要求：有合理的提示，每个功能可以设立菜单，根据提示，可以完成相关的功能要求。

（13）存储结构：根据系统功能要求自行设计，但是要求相关数据要存储在数据文件中。测试数据：要求使用1、全部合法数据；2、局部非法数据。进行程序测试，以保证程序的稳定。

**4、 平衡二叉树编程**

对于1-10000的质数序列<2，3，5，7，…，9973>，建立平衡二叉排序树。

1. 依次查询200-300的每个数是否在树中，将结果写入tree1.txt，格式如下:

XXX no

XXX yes

1. 依次删除500-2000中的每个质数，再查询 600-700之间的每个质数是否在, 将结果写入文件tree2.txt，格式如下:

XXX no

XXX yes

1. 在任务（2）基础上，依次插入1-1000的所有偶数，依次查询100-200的每个偶数是否在树中, 将结果写入文件tree3.txt，格式如下:

XXX no

XXX yes

**5、Huffman编码与解码 (必做)（Huffman编码、二叉树）**

[问题描述]

对一篇不少于10000字符的英文文章（source.txt），统计各字符出现的次数，实现Huffman编码(code.dat)，以及对编码结果的解码(recode.txt)。

[基本要求]

（1） 输出每个字符出现的次数和编码,并存储文件(Huffman.txt)。

（2） 在Huffman编码后，英文文章编码结果保存到文件中(code.dat)，编码结果必须是二进制形式，即0 1的信息用比特位表示，不能用字符’0’和’1’表示。

（3） 实现解码功能。

**6、地铁修建 (必做) （图）**

[问题描述]

　　A市有n个交通枢纽，其中1号和n号非常重要，为了加强运输能力，A市决定在1号到n号枢纽间修建一条地铁。

　　地铁由很多段隧道组成，每段隧道连接两个交通枢纽。经过勘探，有m段隧道作为候选，两个交通枢纽之间最多只有一条候选的隧道，没有隧道两端连接着同一个交通枢纽。

　　现在有n家隧道施工的公司，每段候选的隧道只能由一个公司施工，每家公司施工需要的天数一致。而每家公司最多只能修建一条候选隧道。所有公司同时开始施工。

　　作为项目负责人，你获得了候选隧道的信息，现在你可以按自己的想法选择一部分隧道进行施工，请问修建整条地铁最少需要多少天。

输入格式

　　输入的第一行包含两个整数n, m，用一个空格分隔，分别表示交通枢纽的数量和候选隧道的数量。

第2行到第m+1行，每行包含三个整数a, b, c，表示枢纽a和枢纽b之间可以修建一条隧道，需要的时间为c天。

[基本要求]

输出格式

　　输出一个整数，修建整条地铁线路最少需要的天数。

样例输入

6 6

1 2 4

2 3 4

3 6 7

1 4 2

4 5 5

5 6 6

样例输出

6

样例说明

　　可以修建的线路有两种。

　　第一种经过的枢纽依次为1, 2, 3, 6，所需要的时间分别是4, 4, 7，则整条地铁线需要7天修完；

　　第二种经过的枢纽依次为1, 4, 5, 6，所需要的时间分别是2, 5, 6，则整条地铁线需要6天修完。

第二种方案所用的天数更少。

**7、公交线路提示 (必做) （图）**

[问题描述]

上网下载真实南京公交线路图，建立南京主要公交线路图的存储结构。

[基本要求]

（1）输入任意两站点，给出转车次数最少的乘车路线。

（2）输入任意两站点，给出经过站点最少的乘车路线。

**8、B-树应用 （必做） （查找）**

[问题描述]

设计并实现B-树的动态查找。

[基本要求]

（1） 从文件读取数据

（2）实现B-树的三种基本功能：查找、插入和删除。

（3）以可验证的方式输出结果

**或者**

对1-10000的所有质数，建立m=4的B-tree（每个非叶子结点至少包含1个关键字即2棵子树，最多3个关键字即4棵子树）。

1. 依次查询200-300的每个数是否在B-tree中，将结果写入文件b-tree1.txt，格式如下:

XXX no

XXX yes

1. 依次删除500-2000中的每个质数，再查询 600-700之间的每个质数是否在, 将结果写入文件b-tree2.txt，格式如下:

XXX no

XXX yes

1. 在任务（2）基础上的B-tree，依次插入1-1000的所有偶数，依次查询100-200的每个偶数是否在B-tree中, 将结果写入文件b-tree3.txt，格式如下:

XXX no

XXX yes

**9、排序算法比较 （必做）（排序）**

[问题描述]

利用随机函数产生10个样本，每个样本有100000个随机整数（并使第一个样本是正序，第二个样本是逆序），利用直接插入排序、希尔排序，冒泡排序、快速排序、选择排序、堆排序，归并排序、基数排序8种排序方法进行排序（结果为由小到大的顺序），并统计每一种排序算法对不同样本所耗费的时间。[要求]

(1) 原始数据存在文件中，用相同样本对不同算法进行测试；

(2) 屏幕显示每种排序算法对不同样本所花的时间；

(3) 任意选择1个样本完成数字排序，即统计出每个整数出现的次数，按出现次数从多到少的顺序输出（输出的信息包含数字和出现的频率）。提示：可用一个数组，以下标作为数，数组内容存储该数出现次数来实现（这就相当于直接映射，是机试题目里面常用的一种解题法，很多看似非线性的题型最后其实都可以采取哈希或者映射的方法来巧解，体会哈希思想在机试题目中的巧用）

**（10-20题任选1题）**

【10】 社交网络图中结点的“重要性”计算（选做）（图）

[问题描述]

在社交网络中，个人或单位（结点）之间通过某些关系（边）联系起来。他们受到这些关系的影响，这种影响可以理解为网络中相互连接的结点之间蔓延的一种相互作用，可以增强也可以减弱。而结点根据其所处的位置不同，在网络中体现的重要性也不尽相同。

“紧密度中心性”是用来衡量一个结点达到其他结点的“快慢”的指标，即一个有较高中心性的结点比有较低中心性的结点能够更快地（平均意义下）到达网络中的其他结点，因而在该网络的传播过程中有更重要的价值。在有N个结点的网络中，结点vi的“紧密度中心性” 数学上定义为 到其余所有结点(j!=i)的最短距离的平均值的倒数：



对于非连通图，所有结点的紧密度中心性都是0。

本实验给定一个无权的无向图以及其中的一组结点，要求计算这组结点中每个结点的紧密度中心性。

[基本要求]

（1）输入说明：输入的第一行给出两个正整数N (N<=1000)和M，其中N是图中结点个数，结点编号从1到N； M（M<=10000）是边的数目。随后的M行中，每行给出一条边的信息，即该边连接的两个结点编号，中间用空格分隔。最后一行给出需要计算紧密度中心性的这组结点的个数K（K<=100），以及K个结点的编号，用空格分隔。

（2）输出说明：按照“Cc(i)=x.xx”的格式输出K个给定结点的紧密度中心性，每个输出占一行，结果精确到小数点后2位。

（3）测试用例：

输入 5 8

1 2

1 3

1 4

2 3

3 4

4 5

2 5

3 5

2 4 3

输出 Cc(4) = 0.80

Cc(3) = 1.00

【11】公共钥匙盒（选做）（线性表，栈，队列）

[问题描述]

　　有一个学校的老师共用N个教室，按照规定，所有的钥匙都必须放在公共钥匙盒里，老师不能带钥匙回家。每次老师上课前，都从公共钥匙盒里找到自己上课的教室的钥匙去开门，上完课后，再将钥匙放回到钥匙盒中。

　　钥匙盒一共有N个挂钩，从左到右排成一排，用来挂N个教室的钥匙。一串钥匙没有固定的悬挂位置，但钥匙上有标识，所以老师们不会弄混钥匙。

　　每次取钥匙的时候，老师们都会找到自己所需要的钥匙将其取走，而不会移动其他钥匙。每次还钥匙的时候，还钥匙的老师会找到最左边的空的挂钩，将钥匙挂在这个挂钩上。如果有多位老师还钥匙，则他们按钥匙编号从小到大的顺序还。如果同一时刻既有老师还钥匙又有老师取钥匙，则老师们会先将钥匙全还回去再取出。

　　今天开始的时候钥匙是按编号从小到大的顺序放在钥匙盒里的。有K位老师要上课，给出每位老师所需要的钥匙、开始上课的时间和上课的时长，假设下课时间就是还钥匙时间，请问最终钥匙盒里面钥匙的顺序是怎样的？

[基本要求]

输入格式

　　输入的第一行包含两个整数N, K。

　　接下来K行，每行三个整数w, s, c，分别表示一位老师要使用的钥匙编号、开始上课的时间和上课的时长。可能有多位老师使用同一把钥匙，但是老师使用钥匙的时间不会重叠。

　　保证输入数据满足输入格式，你不用检查数据合法性。

输出格式

　　输出一行，包含N个整数，相邻整数间用一个空格分隔，依次表示每个挂钩上挂的钥匙编号。

样例输入

5 2

4 3 3

2 2 7

样例输出

1 4 3 2 5

样例说明

　　第一位老师从时刻3开始使用4号教室的钥匙，使用3单位时间，所以在时刻6还钥匙。第二位老师从时刻2开始使用钥匙，使用7单位时间，所以在时刻9还钥匙。

　　每个关键时刻后的钥匙状态如下（X表示空）：

　　时刻2后为1X345；

　　时刻3后为1X3X5；

　　时刻6后为143X5；

　　时刻9后为14325。

课程设计要求：

（1）要求从文本文件中输入；

（2）根据时间进程，将取走钥匙和归还钥匙分别视为事件，放入队列中，然后通过每个事件的先后发生对钥匙盒的状态进行变更；

（3）严格按照要求的输入输出格式进行数据的输入、输出（训练CSP考试中的格式化输入输出的正确性）；

（4）选做：通过图形界面来显示钥匙盒的即时状态，以及事件队列的状态。

【12】URL映射（选做）（字符串）

问题描述   
　　URL 映射是诸如 Django、Ruby on Rails 等网页框架 (web frameworks) 的一个重要组件。对于从浏览器发来的 HTTP 请求，URL 映射模块会解析请求中的 URL 地址，并将其分派给相应的处理代码。现在，请你来实现一个简单的 URL 映射功能。   
　　本题中 URL 映射功能的配置由若干条 URL 映射规则组成。当一个请求到达时，URL 映射功能会将请求中的 URL 地址按照配置的先后顺序逐一与这些规则进行匹配。当遇到第一条完全匹配的规则时，匹配成功，得到匹配的规则以及匹配的参数。若不能匹配任何一条规则，则匹配失败。   
　　本题输入的 URL 地址是以斜杠 / 作为分隔符的路径，保证以斜杠开头。其他合法字符还包括大小写英文字母、阿拉伯数字、减号 -、下划线 \_ 和小数点 .。例如，/person/123/ 是一个合法的 URL 地址，而 /person/123? 则不合法（存在不合法的字符问号 ?）。另外，英文字母区分大小写，因此 /case/ 和 /CAse/ 是不同的 URL 地址。   
　　对于 URL 映射规则，同样是以斜杠开始。除了可以是正常的 URL 地址外，还可以包含参数，有以下 3 种：   
　　字符串 ：用于匹配一段字符串，注意字符串里不能包含斜杠。例如，abcde0123。   
　　整数 ：用于匹配一个不带符号的整数，全部由阿拉伯数字组成。例如，01234。   
　　路径 ：用于匹配一段字符串，字符串可以包含斜杠。例如，abcd/0123/。   
　　以上 3 种参数都必须匹配非空的字符串。简便起见，题目规定规则中 和 前面一定是斜杠，后面要么是斜杠，要么是规则的结束（也就是该参数是规则的最后一部分）。而 的前面一定是斜杠，后面一定是规则的结束。无论是 URL 地址还是规则，都不会出现连续的斜杠。   
输入格式   
　　输入第一行是两个正整数 n 和 m，分别表示 URL 映射的规则条数和待处理的 URL 地址个数，中间用一个空格字符分隔。   
　　第 2 行至第 n+1 行按匹配的先后顺序描述 URL 映射规则的配置信息。第 i+1 行包含两个字符串 pi 和 ri，其中 pi 表示 URL 匹配的规则，ri 表示这条 URL 匹配的名字。两个字符串都非空，且不包含空格字符，两者中间用一个空格字符分隔。   
　　第 n+2 行至第 n+m+1 行描述待处理的 URL 地址。第 n+1+i 行包含一个字符串 qi，表示待处理的 URL 地址，字符串中不包含空格字符。

样例输入

5 4  
/articles/2003/ special\_case\_2003  
/articles/<int>/ year\_archive  
/articles/<int>/<int>/ month\_archive  
/articles/<int>/<int>/<str>/ article\_detail  
/static/<path> static\_serve  
/articles/2004/  
/articles/1985/09/aloha/  
/articles/hello/  
/static/js/jquery.js

样例输出

year\_archive 2004  
article\_detail 1985 9 aloha  
404  
static\_serve js/jquery.js

样例说明

　　对于第 1 个地址 /articles/2004/，无法匹配第 1 条规则，可以匹配第 2 条规则，参数为 2004。  
　　对于第 2 个地址 /articles/1985/09/aloha/，只能匹配第 4 条规则，参数依次为 1985、9（已经去掉前导零）和 aloha。  
　　对于第 3 个地址 /articles/hello/，无法匹配任何一条规则。  
　　对于第 4 个地址 /static/js/jquery.js，可以匹配最后一条规则，参数为 js/jquery.js。

数据规模和约定

1 ≤ n ≤ 100，1 ≤ m ≤ 100。  
　　所有输入行的长度不超过 100 个字符（不包含换行符）。  
　　保证输入的规则都是合法的。

【13】行车路线 (选做)（图）

[问题描述]

　　小明和小芳出去乡村玩，小明负责开车，小芳来导航。

　　小芳将可能的道路分为大道和小道。大道比较好走，每走1公里小明会增加1的疲劳度。小道不好走，如果连续走小道，小明的疲劳值会快速增加，连续走s公里小明会增加s2的疲劳度。

　　例如：有5个路口，1号路口到2号路口为小道，2号路口到3号路口为小道，3号路口到4号路口为大道，4号路口到5号路口为小道，相邻路口之间的距离都是2公里。如果小明从1号路口到5号路口，则总疲劳值为(2+2)2+2+22=16+2+4=22。

现在小芳拿到了地图，请帮助她规划一个开车的路线，使得按这个路线开车小明的疲劳度最小。

[基本要求]

输入格式：

　　输入的第一行包含两个整数n, m，分别表示路口的数量和道路的数量。路口由1至n编号，小明需要开车从1号路口到n号路口。

接下来m行描述道路，每行包含四个整数t, a, b, c，表示一条类型为t，连接a与b两个路口，长度为c公里的双向道路。其中t为0表示大道，t为1表示小道。保证1号路口和n号路口是连通的。

输出格式

输出一个整数，表示最优路线下小明的疲劳度。

样例输入

6 7

1 1 2 3

1 2 3 2

0 1 3 30

0 3 4 20

0 4 5 30

1 3 5 6

1 5 6 1

样例输出

76

样例说明

　　从1走小道到2，再走小道到3，疲劳度为52=25；然后从3走大道经过4到达5，疲劳度为20+30=50；最后从5走小道到6，疲劳度为1。总共为76。

课程设计要求：

（1）要求从文本文件中输入；

（2）采用适当的数据结构存储由输入数据中的道路所形成的图结构；

（3）编写尽可能优的算法，处理好连续走小道造成的疲劳值的指数增长（提示：基于迪杰斯特拉算法进行改进即可完成本题）；

（4）除严格按题目要求进行输出以外，还要求输出最优路线的路径，以及从出发点到各个点的最小疲劳值。

【14】算术表达式求值 (选做) （栈）

[问题描述]

　　一个算术表达式是由操作数(operand)、运算符(operator)和界限符(delimiter)组成的。假设操作数是正实数，运算符只含加减乘除等四种运算符，界限符有左右括号和表达式起始、结束符“#”，如：#6+15\*（21-8/4）#。引入表达式起始、结束符是为了方便。编程利用“运算符优先法”求算术表达式的值。

[基本要求]

（1） 从键盘或文件读入一个合法的算术表达式，输出正确的结果。

（2） 显示输入序列和栈的变化过程。

（3） 考虑算法的健壮性，当表达式错误时，要给出错误原因的提示。

（4） 实现非整数的处理（\*）。

【15】编写“连连看”程序。(选做)（图）

[问题描述]

建立一个10\*20的矩形方格图，其中有10种不同的图案，每种图案个数为偶数，填满矩形方格图。

[基本要求]

（1）随机产生原始数据

（2）输入两个位置，如果两者图案相同，并且用少于等于3的直线相连，可消除该两个图案。

【16】Hash表应用 （选做） （查找）

[问题描述]

设计散列表实现VIP客户发掘。对身份证号进行Hash, 通过对乘客某时间段内的乘机频率、里程数统计，发掘VIP客户。

[基本要求]

（1） 设每个记录有下列数据项：身份证号码（虚构，位数和编码规则与真实一致即可）、姓名、航班号、航班日期、里程。

（2） 从文件输入各记录，以身份证号码为关键字建立散列表。

（3） 分别采用开放定址（自行选择和设计定址方案）和链地址两种方案解决冲突；显示发生冲突的次数、每次中解决冲突进行重定位的次数。

（4）记录条数至少在100条以上。

（5） 从记录中实现乘客乘机频率、里程数统计，从而发掘VIP客户。

【17】营业窗口队列模拟 (选做)（队列）

[问题描述]

实现具有n（n=3）个窗口的现实队列模拟，统计每人的等待时间。

[基本要求]

（1） 随机产生顾客的到达时间和服务时间存储文件。

（2） 利用文件数据实现队列的插入和删除。

（3）当有顾客离开时，根据队列长度调整队尾。

（4）考虑顾客中途离队的情况。

（5） 考虑顾客具有优先级的情况。

【18】树的应用 (选做)（树）

[问题描述]

　　JSON (JavaScript Object Notation) 是一种轻量级的数据交换格式，可以用来描述半结构化的数据。JSON 格式中的基本单元是值 (value)，出于简化的目的本题只涉及 2 种类型的值：

　　\* 字符串 (string)：字符串是由双引号 " 括起来的一组字符（可以为空）。如果字符串的内容中出现双引号 "，在双引号前面加反斜杠，也就是用 \" 表示；如果出现反斜杠 \，则用两个反斜杠 \\ 表示。反斜杠后面不能出现 " 和 \ 以外的字符。例如：""、"hello"、"\"\\"。

　　\* 对象 (object)：对象是一组键值对的无序集合（可以为空）。键值对表示对象的属性，键是属性名，值是属性的内容。对象以左花括号 { 开始，右花括号 } 结束，键值对之间以逗号 , 分隔。一个键值对的键和值之间以冒号 : 分隔。键必须是字符串，同一个对象所有键值对的键必须两两都不相同；值可以是字符串，也可以是另一个对象。例如：{}、{"foo": "bar"}、{"Mon": "weekday", "Tue": "weekday", "Sun": "weekend"}。

　　除了字符串内部的位置，其他位置都可以插入一个或多个空格使得 JSON 的呈现更加美观，也可以在一些地方换行，不会影响所表示的数据内容。例如，上面举例的最后一个 JSON 数据也可以写成如下形式。

　　{

　　"Mon": "weekday",

　　"Tue": "weekday",

　　"Sun": "weekend"

　　}

　　给出一个 JSON 格式描述的数据，以及若干查询，编程返回这些查询的结果。

输入格式

　　第一行是两个正整数 n 和 m，分别表示 JSON 数据的行数和查询的个数。

　　接下来 n 行，描述一个 JSON 数据，保证输入是一个合法的 JSON 对象。

　　接下来 m 行，每行描述一个查询。给出要查询的属性名，要求返回对应属性的内容。需要支持多层查询，各层的属性名之间用小数点 . 连接。保证查询的格式都是合法的。

[基本要求]

输出格式

　　对于输入的每一个查询，按顺序输出查询结果，每个结果占一行。

　　如果查询结果是一个字符串，则输出 STRING <string>，其中 <string> 是字符串的值，中间用一个空格分隔。

　　如果查询结果是一个对象，则输出 OBJECT，不需要输出对象的内容。

　　如果查询结果不存在，则输出 NOTEXIST。

样例输入

10 5

{

"firstName": "John",

"lastName": "Smith",

"address": {

"streetAddress": "2ndStreet",

"city": "NewYork",

"state": "NY"

},

"esc\\aped": "\"hello\""

}

firstName

address

address.city

address.postal

esc\aped

样例输出

STRING John

OBJECT

STRING NewYork

NOTEXIST

STRING "hello"

[基本要求]

（1）要求从文本文件中输入；

（2）本题目其实就是一棵普通的树（即每个结点的孩子数不固定，不能单纯采用n叉树来解决），可以考虑使用孩子兄弟表示法等进行表示和存储；

（3）严格按照要求的输入输出格式进行数据的输入、输出（训练CSP考试中的格式化输入输出的正确性）；

（4）选做：使用图形界面（或字符格式化摆成的树形结构，参考Linux下的tree命令），以树状形式显示输入的JSON格式数据。

【19】最小生成树 (选做) （图）

[问题描述]

利用普利姆算法和克鲁斯卡尔算法实现最小生成树问题。

[基本要求]

（1）自行建立图的数据文件，第一行是顶点个数，然后依次是顶点名，接下来是边，用float表示边的权值；

（2）以邻接表或者邻接矩阵表示图皆可；

（3）分别利用Prim和Kruskal算法实现最小生成树；

（4）输出最小生成树的权值之和，及所用的边。

【20】电子小字典（选做）（查找）

[问题描述]

利用键树结构，建立一个微型电子字典。

[基本要求]

实现生词的加入，单词的查找、删除，修改等操作。

其他。。。。（可自选合适的题目，经过教师/助教认定）

成绩评定细则：

* 1. 正确性、功能的完备性和总程序量：程序是否可以运行，结果是否正确，是否实现要求的所有子功能，是否达到足够的程序代码 （50分）

程序代码>=3000 && 必做题 && 选做题 （40分-50分）（优秀）

程序代码>=2500 && 必做题 && 选做题 （30分-39分）（良好）

程序代码>=2000 && 必做题 && 选做题 （20分-29分）（中等）

程序代码>=1500 && 必做题 （10分-19分）（及格）

* 1. 课程设计报告中的算法说明的清晰程度，课程设计报告中总结的深刻程度（20分）
  2. 独立完成情况( 30分）

总计：100分 以五分制（优、良、中、及格、不及格）对应。

4．**根据完成情况，符合要求，提出申请，才可参评优秀。**

加分项目：

1.健壮性：异常处理的情况

2.可读性：代码编写是否规范，是否便于阅读。如函数、变量命名，‘{ }’的缩进，关键位置适量注释等

3.功能的完善：除要求实现的功能外，完成了其它的功能，实现了功能的完善

4.界面的设计：良好交互的界面