**《数据结构与算法》课程设计题目**

**一、上交材料**

课程设计结束后，学生应提交的文档包括**纸质文档和电子文档。纸质文档**包括：

1、《课程设计任务书》一份；

2、《课程设计报告》1份（用塑料拉杆夹装订好）；

纸质文档装入“课程设计资料袋”上交。

**电子文档**包括：

1、课程设计报告

2、课程设计任务书

3、源程序

将这些电子档形成一个压缩文件，文件名为：**第X组-课程设计名**，发送至**指导老师**的邮箱。

**二、评分标准**

百分制，由指导老师根据评分表的具体评价标准进行评分。

## 三、选题要求及课程设计内容

**（一）选题要求：**

1、分组选题，每组3-4人，每组选一题来做。

2、每班不能超过20组，且在同一个班内，每题最多两组参选。

**（二）课程设计内容**

**题目1：城市光缆铺设方案设计**

【问题描述】

在n个城市间建设光纤通信网络，要求仅选择n-1条线路铺设光缆，且达到每个城市都有光缆连通。 由于地理位置的不同，所需经费也不同，采用什么样的设计方案能使总投资最省。

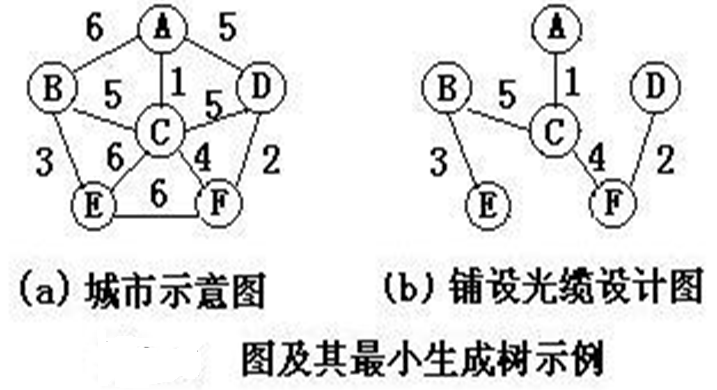
【基本要求】

请用C++语言编写程序，求出以最低经济代价（光缆总量最短）建设这个通信网的方案。城市个数、两个城市间的距离由学生自己设计，存储结构和实现算法由学生自己选定并实现。

具体要求：（1）画出逻辑结构图；（2）画出物理结构图；（3）给出算法设计、实现及时间效率分析。

 【测试数据】

输入城市个数及城市间铺设光缆所需经费，建立至少6个城市之间的数据，以较为直观的方式显示出城市示意图及铺设光缆设计图，如下图所示。



注：图中“顶点”(A、B…)表示城市，顶点之间的连线及其上面的数值表示可以铺设的光缆及所需经费。

【实现提示】

采用最小生成树算法实现。

**题目2：校园地图设计及其应用**

【问题描述】

设计一个广东理工学院校园地图，为来访的客人提供各种信息查询服务。

【基本要求】

(1) 地图所含地点不少于15个。以图中顶点表示校内各地点,存放地点名称、代号、简介等信息;以边表示路径,存放路径长度等相关信息。

(2)设计地图的逻辑结构和物理结构，并分别用图形和C++语言来表示这两种结构。

(3)请用C++语言编写程序，为来访客人提供图中任意地点相关信息的查询，以及为来访客人提供图中任意地点的问路查询,即查询任意两个地点之间的一条最短的简单路径。

(4)给出相应算法的设计、实现及时间效率分析。

【测试数据】

1. 输入广东理工学院各地点名及有关该地点的基本信息，
2. 输入相邻两地点以及它们之间的路径长度。
3. 查询广东理工学院的任一地点的基本信息；
4. 查询广东理工学院任两个地点间的最短路径及路径长度；

【实现提示】

一般情况下，校园的道路是双向通行的，可设校园平面图是一个无向网。顶点和边均含有相关信息。

**题目3：电话号码查询系统的数据结构设计**

【问题描述】

随着通讯电子业务的迅速发展，手机已经进入了家家户户，为了记录手机用户的电话号码，设计一个电话号码查询系统，可完成添加、查找、删除、清空记录、输出电话记录为TXT文档的基本功能。

【基本要求】

1. 设每个记录有下列数据项：电话号码、用户名、地址；
2. 从键盘输入各记录，分别以电话号码和用户名为关键字存储；
3. 存储空间未满时，可以利用空白空间存放记录；
4. 存储电话号码过程时若出现冲突，采用合适的方法解决冲突；
5. 查找并显示给定电话号码的记录；
6. 查找并显示给定用户名的记录。

具体要求：

（1）画出数据结构设计图；（2）给出算法设计、实现及时间效率分析。

 【测试数据】

输入至少约15人以上的电话号码记录，并能实现电话号码的添加、查找、删除、清空等操作。

【实现提示】

1. 设计不同的散列函数，比较冲突率。以电话号码和用户名为关键字存储时，选择冲突率小的哈希函数，建立哈希表进行存放。
2. 在散列函数确定的前提下，存储出现冲突，尝试各种不同类型处理冲突的方法，考察平均查找长度的变化。

**题目4：哈夫曼编译码器**

【问题描述】

利用哈夫曼编码进行信息通信可以大大提高信道利用率，缩短信息传输时间，降低传输成本。但是，这要求在发送端通过一个编码系统对待传数据预先编码，在接收端将传来的数据进行译码（复原）。对于双工信道（即可以双向传输信息的信道），每端都需要一个完整的编/译码系统。试为这样的信息收发站写一个哈夫曼编/译码系统。

【基本要求】

一个完整的系统应具有以下功能：

（1）I：初始化（Initialization）。从终端读入字符集大小n，以及n个字符和n个权值，建立哈夫曼树，并将它存于文件hfmTree中。

（2）E：编码（Encoding）。利用已建好的哈夫曼树（如不在内存，则从文件htmTree中读入），对文件ToBeTran中的正文进行编码，然后将结果存入文件CodeFile中。

（3）D：译码（Decoding）。利用已建好的哈夫曼树将文件CodeFile中的代码进行译码，结果存入文件TextFile中。

（4）P：印代码文件（Print）。将文件CodeFile以紧凑格式显示在终端上，每行50个代码。同时将此字符形式的编码写入文件CodePrint中。

（5）T：印哈夫曼树（Tree Printing）。将已在内存中的哈夫曼树以直观的方式（树或凹入表形式）显示在终端上，同时将此字符形式的哈夫曼树写入文件TreePrint中。

【测试数据】

（1）数据一：已知某系统在通信联络中只可能出现8种字符，其概率分别为0.05,0.29,0.07,0.08,0.14,0.23,0.03,0.11,以此设计哈夫曼编码。利用此数据对程序进行调试。

（2）用下表给出的字符集和频度的实际统计数据建立哈夫曼树，并实现以下报文的编码和译码：“THIS PROGRAM IS MY FAVORITE”。

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 字符 | 空格 | A | B | C | D | E | F | G | H | I | J | K | L | M |
| 频度 | 186 | 64 | 13 | 22 | 32 | 103 | 21 | 15 | 47 | 57 | 1 | 5 | 32 | 20 |
| 字符 | N | O | P | Q | R | S | T | U | V | W | X | Y | Z |  |
| 频度 | 57 | 63 | 15 | 1 | 48 | 51 | 80 | 23 | 8 | 18 | 1 | 16 | 1 |  |

【实现提示】

在程序的一次执行过程中，第一次执行I、D或C命令之后，哈夫曼树已经在内存了，不必再读入。每次执行中不一定执行I命令，因为文件hfmTree可能早已建好。

**题目5：停车场管理方案的数据结构设计**

【问题描述】

停车场是一条可以停放n辆车的狭窄车道，并且只有一个大门是车的出入口。汽车停放按照到达时间的先后顺序依次由东向西排列（大门在最西端，最先到达的第一辆车停在最东端），若停车场已经停满了n辆车，后来的汽车在便道上等候，一旦有车开走，排在便道上的第一辆车可以开入；当停车场的某辆车要离开时，停在他后面的车要先后退为他让路，等它开出后其他车再按照原先次序开入车场，每辆停在停车场的车要按时间长短缴费！

【基本要求】

请用C++语言编写程序实现该停车场的管理过程。

1. 根据车辆到达停车场到车辆离开停车场时所停留的时间进行计时收费。
2. 根据车牌号查到该车辆在停车场或者便道中的位置。
3. 当有车辆从停车场离开时，等待的车辆按顺序进入停车场停放。

 【测试数据】

测试数据：要求使用全部合法数据，整体非法数据，局部非法数据进行程序测试，以保证程序的稳定性。测试数据及其测试结果请在上交的资料文档中写明！

【实现提示】

用栈模拟停车场（后进先出），用队列模拟车场外的便道。按照从终端输入的数据进行模拟管理。数据结构应该包括三个数据项：汽车车牌号码，汽车“到达”或者“离开”信息，汽车到达或者离开的时刻。

**题目6：校园超市选址方案设计**

【问题描述】

现需在广东理工学院设立超市。在校园内，各幢楼到超市的距离不同，同时位于各幢楼的人员去超市的频率也不同。请为超市选址，要求实现总体最优。

【基本要求】

请用C++语言编写程序，实现各幢楼到超市距离、频率最优，确定超市位置，实现总体最优。学校内建筑物数量、两个建筑之间的距离及频率由学生根据广东理工学院实际情况自己设计，存储结构和实现算法由学生自己选定并实现。

具体要求：（1）画出逻辑结构图；（2）画出物理结构图；（3）给出算法设计、实现及时间效率分析。

 【测试数据】

输入建筑物个数、名称，建筑之间的距离、频率，建立至少6幢建筑物间的数据，以较为直观的方式输出各建筑物之间最短路径、距离，并确定超市选址地点。

【实现提示】

可采用Floyd算法或Dijkstra算法实现。

**题目7：表达式的转换及计算**

【问题描述】

程序首先提示输入1或2，若输入1则提示输入一个字符串中缀表达式，求出该表达式的值，并输出该表达式的值及其后缀表达式，若输入的中缀表达式有误，则输出错误提示信息。若输入2则提示输入一个字符串后缀表达式，求出该表达式的值，并输出该表达式的值及其中缀表达式，若输入的后缀表达式有误，则输出错误提示信息。

【基本要求】

请用C++语言编写程序，求出中缀表达式的值。输入的中缀表达式由学生自己输入，存储结构和实现算法由学生自己选定并实现。

具体要求：（1）给出算法的设计、实现原理；（2）给出主要算法的时间效率分析。（3）程序要对错误有良好的提示。

 【测试数据】

输入1，输入中缀表达式：如输入34+45\*10-12#（以#号作为结束符），则输出472，34&45&10&\*&+&12&-，如输入35+=23，则输出“中缀表达式有误，请输入正确的中缀表达式”（以&作为分隔符）；输入2，输入后缀表达式：如输入12&35&10&20&+&-&\*，则输出60，12\*（35-（10+20）），如输入12&35&10&20&+&-&\*&+，则输出“后缀表达式有误，请输入正确的后缀表达式”。

【实现提示】

（1）使用栈来实现。

（2）中缀表达式：我们平时写的表达式：a+b;(a+b)\*c-(a-b)/d

（3）后缀表达式：也叫逆波兰表示法，所有操作符在操作数的后面。上面的中缀表达式对应的后缀表达式ab+;ab+c\*ab-d/-

（4）自定义结束符，分隔符。

**题目8：文件目录结构的显示**

【问题描述】

按照一定的结构，给出操作系统下目录和文件的信息。

【基本要求】

（1）输入要求

输入数据包含几个测试案例。每一个案例由几行组成，每一行都代表了目录树的层次结构。第一行代表目录的根节点。若是目录节点，那么它的孩子节点将在第二行中被输出，同时用一对圆括号“（）”界定。同样，如果这些孩子节点中某一个也是目录的话，那么这个目录所包含的内容将在随后的一行中列出，由一对圆括号将首尾界定。目录的输入格式为：\*nams size，文件的输入格式为：name size，其中\*代表当前节点是目录，name代表文件或目录的名称，由一串长度不大于10的字符组成，并且name字符串中不能含有‘(’，‘)’，‘[’，‘]’和‘\*’。size是该文件/目录的大小，为一个大于0的整数。每一个案例中最多只能包含10层，每一层最多有10个文件/目录。

（2）输出要求

对每一个测试案例，输出时要求：第d层的文件/目录名前面需要插入8\*d个空格，兄弟节点之间要在同一列上。不要使用Tab（制表符）来统一输出的缩进。每一个目录的大小（size）是它所包含的所有子目录和文件大小以及它自身大小的总和。

 【测试数据】

有输入/输出样例如下：

输入样例1：

\*/usr 1

(\*mark 1 \*alex 1)

(hw.c 3 \*course 1)(hw.c 5)

(aa.txt 12)

\*/usr 2

()

输出样例1：

|\_\*/usr[24]

|\_\*mark[17]

| |\_hw.c[3]

| |\_\*course[13]

| |\_aa.txt[12]

|\_\*alex[6]

|\_hw.c[5]

|\_\*/usr[2]

输入样例2：

\*/admin1 1

(\*user1 1 \*user2 1)

(file1.c 3 \*folder1 1) (file2.c 5)

(file3.exe 12)

\*/admin2 1

(\*user3 1 \*user4 1 \*user5 1)

(file4.c 5 \*folder2 1 file6.c 7 file7.com 20) (file8.c 9 \*folder3 1) (file9.avi 100 file10.avi 200)

(file11.exe 10 file12.rm 20) (file13.jpg 10 file14.xls 10)

输入样例2：

|\_\*/admin1[24]

|\_\*user1[17]

| |\_file1.c[3]

| |\_\*folder1[13]

| |\_file3.exe[12]

|\_\*user2[6]

|\_file2.c[5]

|\_\*/admin2[397]

|\_\*user3[64]

| |\_file4.c[5]

| |\_\*folder2[31]

| | |\_file11.exe[10]

| | |\_file12.rm[20]

| |\_file6.c[7]

| |\_file7.com[20]

|\_\*user4[31]

| |\_file8.c[9]

| |\_\*folder3[21]

| |\_file13.jpg[10]

| |\_file14.xls[10]

|\_\*user5[301]

|\_file9.avi[100]

|\_file10.avi[200]

【实现提示】

为了方便对目录的查找、遍历等操作，可以选择孩子兄弟双亲链表来存储树的结构。对目录的大小进行重新计算，根据用户的输入来建立相应的孩子兄弟双亲链表，最后输出树型结构。

步骤： （1）读入给定的目录和文件信息。

（2）建立树型链表以表示目录和文件结构，同时重新计算目录大小

（3）以树型结构输出文件信息。

**题目9：基于约瑟夫环的数字游戏数据结构设计**

【问题描述】

有n个人一起玩游戏，编号分别为1，2… n，并按顺时针方向围坐一圈，每人持有一个密码（正整数）。一开始任选一个正整数作为报数的上限值m，从第一个人开始按顺时针方向自1开始顺序报数，报到m时停止报数，报m的人出列，将他的密码作为新的m值，从他的顺时针方向上的下一个开始重新从1报数，如此下去，直至所有人全部出列为止，最后一个出列的人为最终胜利者。设计一个程序求出出列顺序，得出游戏最终胜利者的编号。

【基本要求】

请用C++语言编写程序，利用单循环链表作为存储结构模拟此过程；键盘输入总人数、初始报数上限值m及各人密码；按照出列顺序输出各人的编号并得出游戏最终胜利者的编号。

具体要求：（1）画出逻辑结构图；（2）画出物理结构图；（3）给出算法设计、实现及时间效率分析。

 【测试数据】

假设有3个人在一起玩游戏（即n=3）,按顺时针坐的依次是编号1、2、3，其对应的密码m分别为3、5、9，并设置初始密码m为8。游戏开始：

2号

1号为起点

3号

（1）分析第一个出列：1、2、3、4、5、6、7、8。

1号为起点

2号

2，5，8

1，4，7

3，6

3号

所以，2号出列，剩下1号、3号，密码m重新设置为2号的密码5。

（2）分析第二个出列：1、2、3、4、5

2，4

1号

2号

1，3，5

3号为起点

所以，3号出列；剩下1号，密码m重新设置为3号的密码9。

（3）分析第三个出列：1、2、3、4、5、6、7、8、9。

2号

1号为起点

1，2，3，4，5，6，7，8，9

3号

因此，出列顺序为：2、3、1，1号为此次游戏的最终胜利者。

【实现提示】

采用单循环链表实现。

**题目10：排队就餐管理方案设计**

【问题描述】

顾客到饭店就餐通常遇到排队等待情况，如果店内有空座，可直接点餐，否则需要排队等待。要求根据顾客的排队情况，及时安排点餐。

【基本要求】

请用C++语言编写程序，模拟顾客排队等待情况。餐桌个数、就餐人数由学生自己设计，存储结构和实现算法由学生自己选定并实现，要求如下。  
（1）饭店内餐桌个数为n个；

（2）顾客到达饭店如果有空座可立即坐下点餐，否则需要依次排队等候；  
（3）一旦有顾客离去，排在队头的顾客便可开始进店点餐；

（4）如果有VIP顾客，可直接插入队头。

程序包含的基本功能说明如下：

1. 排队：输入排队顾客的编号，加入队列；
2. 就餐：排在队列头的顾客进店点餐，并将其从排队队列中删除；
3. 查看排队：从队头到队尾输出所有等待的顾客编号；
4. VIP顾客：直接插入对头；
5. 下班：退出运行，提醒顾客营业结束。

【测试数据】

确定餐桌个数（至少8桌），输入排队顾客编号（不少于10人），以较为直观的方式显示出排队等待的顾客编号及就餐顾客编号。

【实现提示】

采用队列算法实现。