

****

信息技术学院

《程序设计实践》指导书

信息技术学院

2024年6月

**一、设计实践目的**

通过此次实践环节主要达到以下目的：

（1）进一步理解和运用结构化程序设计的思想和方法，学会根据具体问题选择合理的计算机存储结构实现数据的存储，构造较有效率的算法；

（2）学会算法描述的方法，并编制具有结构清晰、合理和易读性的小型实用程序；

（3）并会设计测试方案，完成程序的测试，能撰写出该程序的技术报告，为文档整理工作打下一个初步的基础；

（4）培养小组成员间互相学习，取长补短，协同工作的能力。

**二、设计实践要求**

1．重视设计实践环节，用严谨、科学和踏实的工作态度对待设计实践的每一项任务；

2．按照设计实践的题目要求，独立地完成各项任务，严禁抄袭；凡发现抄袭，抄袭者与被抄袭者皆以零分计入本设计实践成绩。凡发现程序设计实践报告或源程序雷同，涉及的全部人员皆以零分计入本设计实践成绩。

3．认真编写程序设计实践报告。程序设计实践报告的书写格式及内容要求见附录2。

**三、设计实践环境**

1．windows操作系统；

2．Visual C++ 6.0或codeblocks。

**四、设计实践内容及要求**

学生分成若干小组，每个小组有2-3名成员组成，每组一题，由组长主持完成任务的每一个环节。具体设计课题内容见附录1。

**五、设计实践步骤**

1．问题分析和任务定义；

2．数据类型和系统设计；

3．编码实现和静态检查；

4．上机调试；

5．总结和整理程序设计实践报告。

实践环节各阶段时间安排及要求

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 各阶段名称 | 主要内容 | 天数 |
| 0 | 准备阶段 | 通过课程中心网站了解本课程的目标、任务及要求 |  |
| 1 | 任务布置 | 任务布置、小组讨论、制定小组计划 | 1 |
| 2 | 方案确定 | 存储结构及数据类型方案设计、总体程序设计、完成程序结构框图 | 1 |
| 3 | 修正设计方案 | 现场讲评设计方案、修正设计方案 | 1 |
| 4 | 模块算法设计 | 模块算法设计，用流程图描述 | 2 |
| 5 | 上机实践 | 编码、调试、测试 | 3 |
| 6 | 撰写报告 | 撰写程序设计实践报告、汇报作品 | 4 |
| 7 | 上机考核 | 程序现场演示（个人和小组分别进行考核） | 4 |

**六、考核方式和成绩评定**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 总评构成 | 占比 | 考核方式 | 课程目标 | | | | | | 合计 |
| 1 | 2 | 3 | 4 |  |  |
| X1 | 40% | 实践报告 | 30 | 40 | 30 |  |  |  | 100 |
| X2 | 20% | 平时成绩 | 20 | 40 | 20 | 20 |  |  | 100 |
| X3 | 40% | 项目验收 | 20 | 40 | 20 | 20 |  |  | 100 |

考核分为3个部分：

* 项目验收（程序运行、答辩）：按规定时间到机房运行程序，由老师检查运行情况。学生能对自己的程序面对教师提问并能熟练地解释清楚。
* 程序设计实践报告：是否按规定书写程序设计实践报告的各项内容。
* 平时成绩：包括出勤、阶段检查（方案设计验收、汇报，算法设计、编码）、线上学习、课堂互动等。

**七、上交相关内容要求**

1．上交源程序（电子版）：学生按照设计实践的具体要求所开发的所有源程序（应该放到一个文件夹中）；

2．上交程序设计实践报告纸质版和电子版；

3．上交电子版要求：

（1）程序设计实践报告 名字为：程序设计实践报告

（2）源程序（完整）名字为：源程序

（3）将上述两个文件（夹）压缩为一个文件，名字为：程序设计实践报告+组号

**八、参考文献**

1.计算机软件实践教程，郭浩志，西安电子科技大学出版社，1993.12

2.C程序设计教程（第3版），谭浩强，清华大学出版社，2018.8

3.数据结构:用C语言描述（第3版），耿国华，高等教育出版社，2021年7月

**附录1 程序设计实践的具体内容**

# 任务1 学生信息管理系统

在一个数据文件中存放若干学生数据记录，每条记录都有如下数据项：学号，姓名，性别，年龄，成绩。

编一个程序，要有对学生数据的增、删（按学号）、查（分别按学号、姓名）、改（按学号）功能，并能浏览所有学生信息。

要求：表前部为男同学，后部为女同学，并且男女同学都按成绩递减排序（按性别升序排序，同一性别按成绩递减排序），分别计算男生合格率、女生合格率、全班的成绩平均分，并把排序后的学生数据记录及计算结果存入另一个数据文件中。

# 任务2 学生成绩信息管理系统

某班级学生C语言第一次正考的成绩存于数据文件score.txt中，记录了学生学号、姓名和考试成绩，bk.txt文件中记录了补考学生的学号、姓名和补考成绩，编写程序实现以下功能：

（1）具有对学生成绩数据的的增、删（按学号）、查（按学号、按姓名）、改（按学号）功能。

（2）对该班级按成绩降序排序并统计成绩平均分及每个分数段（90-100、80-90、70-80、60-70、60分以下）的人数，结果写入到result.dat中。

（3）再根据补考成绩修改原成绩，修改成绩后仍按成绩降序写在result.dat的后面。成绩修改原则：

①补考成绩大于等于60，原成绩以60分记；②补考成绩小于60，取两次成绩中最高值记。

# 任务3 统计成绩

给出n个学生的m门考试的成绩表，每个学生的信息由学号、姓名以及各科成绩组成。对学生的考试成绩进行有关统计，并打印统计表。具体功能如下：

1. 具有对学生信息的增、删（按学号）、查（按学号、姓名）、改（按学号）功能。

　　（2） 按总分高低次序，打印出名次表，总分相同的为同一名次；

（3） 按名次打印出每个学生的学号、姓名、总分以及各科成绩。

（4） 输出每科课程的平均成绩。

# （5）保存：数据永久性存放至外存。

# （6）读取：将存放于外存的数据读入内存继续操作。

# 任务4 图书信息管理系统

在一个数据文件中存放若干图书数据记录，每条记录都有如下数据项：书名，作者，出版社，ISBN，出版时间，价格，类别。

编写一个程序，具有对图书数据的的增、删（按ISBN）、查（分别按类别、出版社、书名、作者）、改（按ISBN）功能，并能浏览全部图书信息。

要求：按出版社进行分类存取图书信息，同一出版社中的图书信息按照出版时间降序排列（按出版社升序排序，同一出版社中的图书信息按照出版时间降序排序）。分别统计每个出版社出版的图书数量，并把排序后的图书记录及计算结果存入另一个数据文件中。

# 任务5 旅游信息管理系统

在一个数据文件中存放若干记录，每条记录都有如下数据项：线路编号、目的地、旅游天数、旅游类别、出行月份、价格。

编写一个程序，具有对旅游信息数据的增、删（按线路编号）、查（分别按线路编号、目的地、旅游类别）、改（按线路编号）功能，并能浏览全部旅游信息。要求：按旅游目的地进行分类存取，同一目的地的旅游线路按照月份升序排序（按旅游目的地升序排序，同一目的地的旅游线路按照月份升序排序）。分别统计每个月、每个目的地的旅游线路总数。把排序后的旅游记录及旅游线路总数存入另一个数据文件中。

# 任务6 学生宿舍信息管理系统

已知宿舍的信息包括：宿舍楼号，宿舍号，床位号，对应床位号的学生学号，楼长姓名等。试设计程序实现以下功能：

（1）宿舍信息录入：可随时增加宿舍信息到数据文件中。

（2）宿舍信息浏览：宿舍的信息输出到屏幕。

（3）查询：按学号查询学生所住宿舍楼号、宿舍号等信息，按宿舍楼号查询出该楼所有空床位的信息。

（4）修改：修改指定宿舍的信息。

（5）删除：删除指定宿舍的信息。

（6）统计：计算指定宿舍楼的宿舍床位的空床数。

（7）排序：按学生学号进行升序排序。

# （8）保存：数据永久性存放至外存。

# （9）读取：将存放于外存的数据读入内存继续操作。

# 任务7 跟踪管理系统

# 要对某人某天所到之处进行跟踪，记录其按时间先后顺序所到过的地名、地址、开始时间、结束时间、事务等信息。试设计程序实现以下功能：

# （1）添加：发现跟踪对象到了一个新地方，需要记录下来，添加到前一个地名的后面。

# （2）按顺序输出：按顺序输出跟踪对象全天所到过的地方。

# （3）查询：按地名查询此人是否到过某地方，到过的话输出相关信息，并标明是到过的

# 第几个地方。

# （4）删除：发现在记录过程中把某个本没有到过的地方给错误地添加进去了，进行删除。

# （5）插入：发现某个到过的地方没有记录进去，需插入到合适的位置。

# （6）修改：某个到过的地方记录有误，需要重新更正。

（7）排序：按所到过地名按字母顺序升序排序。

（8）统计：统计到过某一地点的次数。

# （9）保存：数据永久性存放至外存。

# （10）读取：将存放于外存的数据读入内存继续操作。

# 任务8 某航空公司飞机购票模拟系统（服务器端）

文件信息：

航线信息：航线编码、起点、终点、公里数、飞行时间。

航班信息：航班编号、航线编码、起飞时间、可售机票总数、票价、当前状态（准备、已出票、售完、已结束）。

售票信息：航班编号、顾客ID、顾客姓名、购票时间（y-m-d-h-m-s）。

设计程序能实现以下航空公司服务器端功能：

（1）航线的维护（创建、修改（按航线编号）、删除（按航线编号，不能删除已出票、售完航班））

（2) 航班的维护（创建、修改（按航班编号）、删除（按航班编号，不能删除已出票、售完航班））。

（3）显示航班列表。

（4）显示指定航班的售票情况（明细和统计值）。

（5）显示各航班的销售情况。

（6）按起飞时间对航班信息进行升序排序。

# （7）保存：数据永久性存放至外存。

# （8）读取：将存放于外存的数据读入内存继续操作。

# 任务9 某航空公司飞机购票模拟系统（客户端）

文件信息：

航线信息：航线编码、起点、终点、公里数、飞行时间。

航班信息：航班编号、航线编码、起飞时间、可售机票总数、票价、当前状态（准备、已出票、售完、已结束）。

售票信息：航班编号、顾客ID、顾客姓名、购票时间（y-m-d-h-m-s）。

设计程序能实现以下顾客端功能：

（1）搜索指定航线、指定时间段的航班。

（2）购票。

（3）退票。

（4）按顾客ID修改售票信息。

（5）输入顾客ID，显示顾客的购票信息。

（6）根据售票信息统计指定航班的售票情况。

（7）按航班号从小到大显示详细售票信息。

# （8）保存：数据永久性存放至外存。

# （9）读取：将存放于外存的数据读入内存继续操作。

# 任务10 简易车票信息管理系统

一车站每天有n个发车班次，每个班次都有一班次号（1、2、3…n），固定的发车时间，固定的路线（起始站、终点站），大致的行车时间，固定的额定载客量。如：

班次 发车时间 起点站 终点站 行车时间 额定载量 已定票人数

1 8:00 上海 杭州 2 45 30

2 6:30 上海 上虞 3 40 40

3 7:00 上海 上虞 3 40 20

4 10:00 上海 上虞 3 40 2

…

设计程序能实现以下功能：

（1）录入各班次信息(信息用文件保存),可不定时地增加班次数据；

（2）按时间从小到大次序浏览班次信息,可显示出所有班次当前状态（如果当前系统时间超过了某班次的发车时间,则显示“此班已发出”的提示信息)；

（3）按班次号修改班次信息。

（4）按班次号删除班次信息。

（5）查询路线：可按班次号查询 ,可按终点站查询（结果按班次时间排序）；

（6）售票和退票功能

A：当查询出已定票人数小于额定载量且当前系统时间小于发车时间时才能售票，自动更新已售票人数

B：退票时，输入退票的班次，当本班车未发出时才能退票，自动更新已售票人数

要求初始有文本菜单可供选择相应功能。

# （7）保存：数据永久性存放至外存。

（8）读取：将存放于外存的数据读入内存继续操作。

# 任务11简单的学籍管理系统

使用下面的数据，设计一个简单的学籍管理系统，实现出最基本的功能。

学生基本信息文件(A.TXT)及其内容：A.TXT文件不需要编程录入数据，可用文本编辑工具直接生成：

学号 姓名 性别 宿舍号码 电话号码

01 张成成 男 501 87732111

02 李成华 女 101 87723112

03 王成凤 女 101 87723112

04 陈东 男 501 87732111

… … … … ………

学生成绩基本信息文件(B.TXT)及其内容：不需要编程录入数据，可用文本编辑工具直接生成每个学生的学号、课程编号、课程名称、学分、平时成绩、实验成绩、卷面成绩共7个数据，综合成绩、实得学分由程序根据条件自动运算。

学号 课程编号 课程名称 学分 平时成绩 实验成绩 卷面成绩 综合成绩 实得学分

01 A01 大学物理 3 66 78 82

02 B03 高等数学 4 78 -1 90

01 B03 高等数学 4 45 -1 88

02 C01 C语言 3 65 76 66

… … ……… … … … …

设计程序能实现以下功能：

（1）数据初始化功能: B.TXT中综合成绩、实得学分由程序根据条件自动运算

综合成绩的计算：如果本课程的实验成绩为-1，则表示无实验，综合成绩=平时成绩\*30%+卷面成绩\*70%; 如果实验成绩不为-1,表示本课程有实验,综合成绩=平时成绩\*15%+实验成绩\*15%+卷面成绩\*70% 。

实得学分的计算: 采用等级学分制：

综合成绩在90-100之间 ,应得学分=学分\*100% ；综合成绩在80-90之间 ,应得学分=学分\*80%；综合成绩在70-80之间 ,应得学分=学分\*75%；综合成绩在60-70之间 ,应得学分=学分\*60%；综合成绩在60以下 ,应得学分=学分\*0%

（2）查询功能：分为学生基本情况查询和成绩查询两种：

A：学生基本情况查询：

A1----输入一个学号或姓名（可实现选择），查出此生的基本信息并显示输出；

A2---输入一个宿舍号码，可查询出本室所有的学生的基本信息并显示输出。

B：成绩查询：

B1：输入一个学号时，查询出此生的所有课程情况，格式如下：

学 号：xx 姓 名：xxxxx

课程编号：xxx 课程名称：xxxxx 综合成绩：xxxx 实得学分: xx

课程编号：xxx 课程名称：xxxxx 综合成绩：xxxx 实得学分: xx

…

共修：xx科，实得总学分为: xxx

（3）删除功能：当在A.TXT中删除一个学生时，自动地在B.TXT中删除此人所有信息。

（4）录入学生基本信息并存储在文件中。

（5）按学号修改学生基本信息并存储在文件中。

（6）排序功能：能实现选择按综合成绩或实得学分升序或降序排序并显示数据。

要求初始有文本菜单可供选择相应功能。

# 任务12 学生奖惩信息管理系统

学生奖惩的信息包括：编号、学号、姓名、班级、学院、类别、级别、名称、详细内容、日期、授予单位等。系统能够完成学生奖惩信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按学号对所有学生奖惩的信息进行排序。

（2） 查询：分别按学号、班级和学院查找学生的奖惩信息。

（3） 更新：按编号对某个学生奖惩的某项信息进行修改。

（4） 插入：加入新学生奖惩的信息。

（5） 删除：按学号删除已退学的学生奖惩的信息。

（6） 浏览：输出所有学生奖惩信息。

（7） 统计：根据输入类别和级别，统计该类别和级别的奖惩人数并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

任务13 用电信息管理系统

用电的信息包括：编号、企业编号、企业名称、联系电话、谷电量、峰电量、谷价、峰价、年月、合计电量、合计电费等。系统能够完成用电情况信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按企业编号对所有用电情况的信息进行排序。

（2） 查询：分别按企业编号和企业名称查找该企业某月的用电情况。

（3） 更新：按编号对某个用电情况的某项信息进行修改。

（4） 插入：加入新用电情况的信息。

（5） 删除：按企业编号删除已注销企业的用电情况信息。

（6） 浏览：输出所有用电情况信息。

（7） 统计：统计某月和某年的合计电量和合计电费总量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务14 设备采购信息管理系统

设备采购的信息包括：编号、设备编号、设备名称、生产单位、单价、部门编号、部门名称、数量、要求到货时间、联系人等。系统能够完成设备采购信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按设备编号对所有设备采购的信息进行排序。

（2） 查询：分别按编号、设备编号和设备名称查找设备采购信息。

（3） 更新：按编号对某个设备采购的某项信息进行修改。

（4） 插入：加入新设备采购的信息。

（5） 删除：按设备编号删除已淘汰设备的采购信息。

（6） 浏览：输出所有设备采购信息。

（7） 统计：统计某部门采购某种设备的总数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务15 商品销售信息管理系统

商品销售的信息包括：流水号、商品编号、商品名称、营业员编号、姓名、数量、单价、小计、结算时间等。系统能够完成商品销售信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按商品编号对所有商品销售的信息进行排序。

（2） 查询：分别按商品编号、商品名称和营业员编号查找商品销售信息。

（3） 更新：按流水号对某个商品销售的某项信息进行修改。

（4） 插入：加入新商品销售的信息。

（5） 删除：按营业员编号删除已离职营业员的商品销售信息。

（6） 浏览：输出所有商品销售信息。

（7） 统计：统计某营业员销售某种商品的总的数量和总的金额并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务16 学生考评信息管理系统

学生考评的信息包括：学号、姓名、性别、出生日期、专业、班级、所属学院、德育考评分、体育考评分、智育考评分、综合考评分等，综合分=0.7\*智育分+0.2\*德育分+0.1\*体育分。系统能够完成学生考评信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按学号对所有学生考评的信息进行排序。

（2） 查询：分别按学号、姓名、班级查找学生考评信息。

（3） 更新：按学号对某个学生考评的某项信息进行修改。

（4） 插入：加入新学生考评的信息。

（5） 删除：按班级删除已毕业学生的考评信息。

（6） 浏览：输出所有学生考评信息。

（7） 统计：统计某学院的某班级综合分不及格的学生人数并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务17 图书借阅信息管理系统

图书借阅的信息包括：编号、图书编号、书名、出版社、学号、姓名、借出时间、归还时间、期限等。系统能够完成借阅信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按图书编号对所有借阅的信息进行排序。

（2） 查询：分别按图书编号、书名、学号查找图书借阅信息。

（3） 更新：按编号对某个借阅信息的某项信息进行修改。

（4） 插入：加入新的图书借阅信息。

（5） 删除：按图书编号删除已下架图书的借阅信息。

（6） 浏览：输出所有借阅信息。

（7） 统计：统计某出版社某种图书借阅的次数并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务18 人事信息管理系统

人员的信息包括：人员编号、姓名、性别、出生日期、联系电话、最高学历、职称、户口所在地等。系统能够完成人员信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按人员编号对所有人员的信息进行排序。

（2） 查询：分别按人员编号、姓名、职称查找人员信息。

（3） 更新：按编号对某个人员的某项信息进行修改。

（4） 插入：加入新人员的信息。

（5） 删除：按编号删除已离职的人员的信息。

（6） 浏览：输出所有人员信息。

（7） 统计：统计某职称某最高学历的人数并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务19 员工信息管理系统

员工的信息包括：编号、姓名、性别、出生年月、学历、职务、电话、住址等。系统能够完成员工信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有员工的信息进行排序。

（2） 查询：分别按编号、姓名、学历条件查找员工信息。

（3） 更新：按编号对某个员工的某项信息进行修改。

（4） 插入：加入新员工的信息。

（5） 删除：按编号删除已离职的员工的信息。

（6） 浏览：输出所有员工信息。

（7） 统计：统计某学历某职务的员工人数并输出。

# （8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务20 简单个人电话号码查询系统

**1) 问题描述**

人们在日常生活中经常需要查找某个人或某个单位的电话号码，本实验将实现一个简单的个人电话号码查询系统，电话号码信息至少包含4个字段（如号码、姓名、分类、地址等），根据用户输入的信息（例如姓名等）进行快速查询。

**2) 基本要求**

(1) 在外存上，用文件保存电话号码信息；

(2) 在内存中，设计数据结构存储电话号码信息；

(3) 提供查询功能：根据电话号码、姓名实现快速查询；

(4) 能够浏览全部电话号码信息。

(5) 提供其他维护功能：例如插入、删除（按电话号码、姓名）、修改（按电话号码）等；

(6) 按电话号码进行排序。

(7) 按分类统计每种类型电话号码的数量。

# 任务21 学分管理程序

**1) 问题描述**

请设计一个学生的学分管理程序。

假设每位学生必须完成基础课50学分、专业课50学分、选修课24学分、人文类课程8学分、实验性课程20学分才能够毕业。因此在管理学分时，要考虑每个学分所属于的课程类别。学分信息应该包括学号、姓名、班号、课程类别、学分等。

**2) 基本要求**

该程序应该具有下列功能：

（1） 通过键盘输入某位学生的学分信息；

（2）给定学号，显示某位学生的学分完成情况；

（3） 给定某个班级的班号，显示该班所有学生学分完成情况；

（4） 给定某位学生的学号，修改该学生的学分信息；

（5） 按照某类课程的学分高低进行排序；

（6） 给定某位学生的学号，删除该学生的学分信息；

（7） 提供一些统计各类信息的功能（如输出能够毕业的学生人数及明细；某类课程学分完成人数及明细）。

# （8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务22 销售管理系统

**1)问题描述**

 某公司有四个销售员（编号：1-4），负责销售五种产品（编号：1-5）。每个销售员都将当天出售的每种产品各写一张便条交上来。每张便条包含内容：

（1）便条编号

（2）销售员的代号

（3）产品的代号

（4）这种产品的当天的销售额

（5）日期

每位销售员每天可能上缴0-5张便条。假设，收集到了上个月的所有便条，编写一个处理系统，读取上个月的销售情况（自己设定），进行如下处理。

**2) 基本要求**

（1）对便条具有增、删（按便条编号）、查（按便条编号、销售员的代号、产品的代号、日期、全部）、改（按便条编号）功能。

（2）计算上个月每个人每种产品的销售额。

（3）按销售额对销售员进行排序，输出排序结果（销售员代号和销售额）

（4）统计每种产品的总销售额，对这些产品按从高到低的顺序，输出排序结果（需输出产品的代号和销售额）

# （5）保存：数据永久性存放至外存。

（6）读取：将存放于外存的数据读入内存继续操作。

# 任务23 库存管理系统

**1)问题描述**

试设计一库存管理系统，产品信息包括产品编号、名称、型号、厂家、价格、数量等（产品编号不重复）。

**2) 基本要求**

该系统应具有以下功能：

（1）产品信息录入功能(产品信息用文件保存)－－输入

（2）产品信息浏览功能 －－输出

（3）产品入库

（4）产品出库

（5）查询和排序功能：

 1）按价格从大到小排序

 2）按名称查询

（6）产品信息删除（按产品编号）、修改功能（按产品编号）。

（7）统计功能：查询某种型号产品的库存数量并输出。

# （8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务24 学生作业完成情况管理程序

**1)问题描述**

请设计一个学生作业完成情况管理程序。

假设某门课程一学期要留10次作业，每次老师要进行批改，给出分数后还要进行登记。学期期末要根据每次作业的成绩计算出最终的平时成绩（满分100）。作业登记信息应该包含：学号、姓名、10次作业的完成情况。

**2) 基本要求**

该程序应该具有下列功能：

（1） 通过键盘输入某位学生某次作业的分数（每次录入一位学生的一次作业分数）；

（2） 给定学号，显示某位学生作业完成情况；

（3） 给定某个班级的班号，显示该班所有学生的作业完成情况；

（4） 给定某位学生的学号，修改该学生的作业完成信息；

（5）定某位学生的学号，删除该学生的信息；

（6） 按学生的最终平时成绩进行排序；

（7） 输出平均分数。

# （8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务25 学生籍贯管理系统

**1)问题描述**

编制一个学生籍贯信息记录簿，每个学生信息包括：学号、姓名、籍贯、班级、联系电话、通信地址。

**2) 基本要求**

该程序应该具有下列功能：

（1）输入学生信息并以磁盘文件保存；

（2）读取磁盘文件并显示输出所有学生的籍贯信息；

（3）按学号或姓名查询其籍贯；

（4）按籍贯查询并输出该籍贯的所有学生；

（5）能添加、删除（按学号）和修改（按学号）学生的籍贯信息；

（6）显示输出上海籍和非上海籍学生的信息并可分别存盘；

（7）浏览所有学生的籍贯信息；

（8） 按学号进行排序。

# 任务26 机票管理系统

**1)问题描述**

一机场每天有n个航班，每个班次都有一班次号（1、2、3…n），固定的起飞时间，固定的路线（起始站、终点站），大致的飞行车时间，固定的额定载客量。如

班次 起飞时间 起点站 终点站 飞行时间 额定载量 已定票人数

1 8:00 上海 广汉 2 145 130

2 6:30 上海 成都 0.5 140 140

3 7:00 上海 成都 0.5 140 120

4 10:00 上海 成都 0.5 140 120

…

试设计一个机票管理系统，对机场的售票情况进行管理。

**2) 基本要求**

功能要求：

（1）录入班次信息(信息用文件保存),可不定时地增加班次数据；

（2） 按时间从小到大次序浏览班次信息，可显示出所有班次当前状况(如果当前系统时间超过了某班次的起飞时间,则显示“此班已发出”的提示信息)。

（3）查询路线：可按班次号查询 ,可按终点站查询；

（4）按班次修改班次信息。

（5）按班次删除班次信息。

（6）售票和退票功能

A：当查询出已定票人数小于额定载量且当前系统时间小于起飞时间时才能售票，自动更新已售票人数

B：退票时，输入退票的班次，当本航班飞机未发出时才能退票，自动更新已售票人数。

# （7）保存：数据永久性存放至外存。

（8）读取：将存放于外存的数据读入内存继续操作。

# 任务27 通信录管理系统

**1)问题描述**

设计出模拟手机通信录管理系统，实现对手机中的通信录进行管理。

**2) 基本要求**

功能要求：

* 1. 查看功能：选择此功能时，列出下列三类选择。

A 工作类 B 家庭类 C 朋友类 ，当选中某类时，显示出此类所有数据中的姓名和电话号码）

* 1. 增加功能：能录入新数据（一个结点包括：姓名、电话号码、分类（可选项有：A 工作类 B 家庭类 C 朋友类）、家庭住址）。例如

杨春 13589664454 工作类 静安区

当录入了重复的姓名和电话号码时，则提示数据录入重复并取消录入；当通信录中超过100条信息时，存储空间已满，不能再录入新数据；录入的新数据能按递增的顺序自动进行条目编号。

（3）修改功能：选中某个人的姓名时，可对此人的相应数据进行修改

（4）删除功能：选中某个人的姓名时，可对此人的相应数据进行删除，并自动调整后续条目的编号。

（5）能按编号从低到高进行排序。

（6）能统计通讯录中不同类别的记录个数。

（7）保存：数据永久性存放至外存。

（8）读取：将存放于外存的数据读入内存继续操作。

# 任务28 成绩管理系统

**1）问题描述**

现有学生成绩信息，内容如下：

姓名 学号 语文 数学 英语

张明明 01 67 78 82

李成友 02 78 91 88

张辉灿 03 68 82 56

王露 04 56 45 77

陈东明 05 67 38 47

…. .. .. .. …

请编写一系统，实现学生信息管理。

**2) 基本要求**

功能要求：

1. 信息维护：

要求：学生信息数据要以文件的形式保存，能实现学生信息数据的维护。此模块包括子模块有：增加学生信息、删除学生信息（按学号）、修改学生信息（按学号）。

1. 信息查询：

要求：查询时可实现按姓名查询、按学号查询、浏览全部信息

1. 成绩统计：

要求：A输入任意的一个课程名（如数学）和一个分数段（如60--70），统计出在此分数段的学生情况。

（4）排序：能对用户指定的任意课程名，按成绩升序或降序排列学生数据并显示排序结果（使用表格的形式显示排序后的输出结果）（使用多种方法排序者，加分）

# 任务29 图书管理系统

**1)问题描述**

设计一个系统，对图书信息进行管理，信息描述：有关该系统基本信息的描述，如：图书名称、图书编号、单价、作者、存在状态、借书人姓名、性别、学号等。

**2) 基本要求**

基本功能：

（1）新进图书基本信息的输入。

（2）图书基本信息的浏览。

（3）对撤消图书信息的删除。

（4）按图书编号查询图书信息。

（5）办理借书手续（非注册会员不能借书）。

（6）办理还书手续。

（7）统计图书库存、已借出图书数量。

（8）按图书编号从小到大顺序输出所有图书信息。

（9）保存：数据永久性存放至外存。

（10）读取：将存放于外存的数据读入内存继续操作。

# 任务30 商店存货管理系统

**1)问题描述**

试设计一商店存货管理系统，商品信息包括商品编号、名称、价格、数量、进货时间、保质期等（商品编号不重复）。要求每次出货时取进货时间最早且最接近保质期中止时间的货物。

**2) 基本要求**

该系统应具有以下功能：

（1）商品信息录入功能(商品信息用文件保存)－－输入

（2）商品信息浏览功能 －－输出

（3）商品入库

（4）商品出库

（5）查询和排序功能：

 1）按价格从大到小排序

 2）按名称查询

3）按编号查询

1. 商品信息删除（按商品编号）、修改（按商品编号）功能。

# 任务31 个人帐簿管理系统设计

个人帐簿管理系统记录和管理某人每月的全部收入及各项开支情况，包括食品消费，房租，子女教育费用，水电费，医疗费，储蓄等。

具体要求如下：进入系统后可以输入和修改某月的收支情况，可以对每月的开支从小到大进行排序，可以根据输入的月份查询每月的收支情况。

（1）可以输入、修改和删除某月的收支情况，并以文件保存个人的收支情况。

（2）可以根据输入的月份分别查询每月的收入和开支情况。

（3）可以对每月的开支按日期从小到大进行排序。

（4）根据输入月份统计该月的总收入和总支出。

# 任务32 宿舍管理系统

**1)问题描述**

设某宿舍有:101,102,201,202四个房间,每个房间有4个床位,学生信息包括学号、姓名、房间号、床号，为学生宿舍管理人员编写一个宿舍管理软件。

**2) 基本要求**

该程序应该具有下列功能：

（1）学生的入住处理；

（2）学生退房处理；

（3）输出学生入住信息(按房间号和床号有序)；

（4）修改入住信息（按学号）；

（5）学生调换宿舍或床位处理；

（6）按给定学号、姓名、房号查询学生入住信息；

（7）查询房间使用情况。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务33 车辆信息管理系统

车辆的信息包括：车辆编号、车主姓名、车牌号、型号、联系电话、住址等。系统能够完成车辆信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按车辆编号对所有车辆的信息进行排序。

（2） 查询：分别按车辆编号、车主姓名条件查找车辆信息。

（3） 更新：按车辆编号对某个车辆的某项信息进行修改。

（4） 插入：加入新车辆的信息。

（5） 删除：按车辆编号删除报废的车辆信息。

（6） 浏览：输出所有车辆信息。

（7） 统计：统计某车主的车辆数量。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务34 课程信息管理系统

课程的信息包括：课程编号、课程名称、性质、学分、开课学期、类别、开课单位等。系统能够完成课程信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按课程编号对所有课程的信息进行排序。

（2） 查询：分别按课程编号、课程名称、课程性质条件查找课程信息。

（3） 更新：按课程编号对某门课程的某项信息进行修改。

（4） 插入：加入新课程的信息。

（5） 删除：按编号删除已不开设课程的信息。

（6） 浏览：输出所有课程信息。

（7） 统计：统计某开课单位某学期开课的门数并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务35 机房信息管理系统

机房的信息包括：电脑编号、网络编号、型号、购买日期、供应商、状态、维修情况、所在机房编号等。系统能够完成电脑信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按电脑编号对所有电脑的信息进行排序。

（2） 查询：分别按电脑编号、型号、购买日期条件查找电脑信息。

（3） 更新：按电脑编号对某个电脑的某项信息进行修改。

（4） 插入：加入新电脑的信息。

（5） 删除：按电脑编号删除已报废的电脑的信息。

（6） 浏览：输出所有电脑信息。

（7） 统计：统计某机房某型号电脑的数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务36 学生考勤信息管理系统

学生考勤的信息包括：编号、学号、姓名、性别、班级、日期、节次、出勤情况、备注等。系统能够完成学生出勤信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按学号对所有学生的出勤信息进行排序。

（2） 查询：分别按编号、学号、姓名条件查找学生的出勤信息。

（3） 更新：按编号对某个学生的某项出勤信息进行修改。

（4） 插入：加入新出勤的信息。

（5） 删除：按学号删除已退学的学生出勤的信息。

（6） 浏览：输出所有学生出勤信息。

（7） 统计：统计某学生在某月内的出勤和缺勤天数并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务37 教工考勤信息管理系统

教工的出勤信息包括：编号、教师号、姓名、年龄、学历、职称、电话、日期、出勤信息等。系统能够完成出勤信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按教师号对所有教工的出勤信息进行排序。

（2） 查询：分别按编号、教师号、姓名条件查找教工的出勤信息。

（3） 更新：按编号对某个教工的某项出勤信息进行修改。

（4） 插入：加入新教工的出勤信息。

（5） 删除：按编号删除已离职的教工的出勤信息。

（6） 浏览：输出所有教工的出勤信息。

（7） 统计某教师某月的出勤和缺勤天数并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务38 药品信息管理系统

药品的信息包括：药品编号、名称、类型、批号、有效期、生产厂家、购买日期、数量等。系统能够完成药品信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按药品编号对所有药品的信息进行排序。

（2） 查询：分别按药品编号、名称、生产厂家条件查找药品信息。

（3） 更新：按药品编号对某个药品的某项信息进行修改。

（4） 插入：加入新药品的信息。

（5） 删除：按药品编号删除已损耗的药品的信息。

（6） 浏览：输出所有药品信息。

（7） 统计：统计某生产厂家某类型药品的总量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务39 体育器材管理系统

体育器材的信息包括：编号、名称、单价、类别、状态、生产厂家、购买日期等。系统能够完成体育器材信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有体育器材的信息进行排序。

（2） 查询：分别按编号、名称、类别条件查找体育器材信息。

（3） 更新：按编号对某个体育器材的某项信息进行修改。

（4） 插入：加入新体育器材的信息。

（5） 删除：按编号删除已报废的体育器材的信息。

（6） 浏览：输出所有体育器材信息。

（7） 统计：输出某生产厂家某类别体育器材的数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务40 毕业生就业情况管理系统

毕业生就业情况的信息包括：学号、姓名、性别、年龄、班级、专业、就业情况、备注等。系统能够完成毕业生就业信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按学号对所有毕业生就业的信息进行排序。

（2） 查询：分别按学号、姓名、班级条件查找毕业生就业情况。

（3） 更新：按学号对某个毕业生就业的某项信息进行修改。

（4） 插入：加入新毕业生就业的信息。

（5） 删除：按学号删除已退学的毕业生就业的信息。

（6） 浏览：输出所有毕业生就业信息。

（7） 统计：统计某专业某班级毕业生就业数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务41 房屋信息管理系统

房屋的信息包括：编号、位置、产权性质、产权年限、建筑日期、类型、建筑面积、使用面积、状态等。系统能够完成房屋信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有房屋的信息进行排序。

（2） 查询：分别按编号、类型、建筑面积条件查找房屋信息。

（3） 更新：按编号对某个房屋的某项信息进行修改。

（4） 插入：加入新房屋的信息。

（5） 删除：按编号删除已注销的房屋信息。

（6） 浏览：输出所有房屋信息。

（7） 统计某类型某使用面积的房屋套数并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务42 房屋销售管理系统

房屋销售的信息包括：编号、位置、产权性质、产权年限、建筑日期、类型、建筑面积、使用面积、售价、销售情况、销售日期等。系统能够完成房屋信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有房屋销售的信息进行排序。

（2） 查询：分别按编号、类型、售价条件查找房屋销售信息。

（3） 更新：按编号对某个房屋销售的某项信息进行修改。

（4） 插入：加入新房屋销售的信息。

（5） 删除：按编号删除已注销的房屋销售信息。

（6） 浏览：输出所有房屋销售信息。

（7） 统计：统计已销售的某类型某产权年限房屋的销售总额并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务43 房屋租赁管理系统

房屋租赁的信息包括：编号、位置、类型、使用面积、价格、租赁情况、租赁日期、租赁时长等。系统能够完成房屋信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有房屋租赁的信息进行排序。

（2） 查询：分别按位置、使用面积、价格条件查找房屋租赁信息。

（3） 更新：按编号对某个房屋租赁的某项信息进行修改。

（4） 插入：加入新房屋租赁的信息。

（5） 删除：按编号删除已注销的房屋租赁信息。

（6） 浏览：输出所有房屋租赁信息。

（7） 统计：统计已租赁的某类型某使用面积房屋的租赁价格总额并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务44 计算机销售管理系统

计算机销售的信息包括：电脑编号、型号、生厂商、销售日期、售价、客户姓名、联系方式等。系统能够完成电脑信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按电脑编号对所有计算机销售的信息进行排序。

（2） 查询：分别按电脑编号、型号、客户姓名条件查找计算机销售信息。

（3） 更新：按电脑编号对某个计算机销售的某项信息进行修改。

（4） 插入：加入新计算机销售的信息。

（5） 删除：按电脑编号删除已注销的计算机销售信息。

（6） 浏览：输出所有计算机销售信息。

（7） 统计：统计某型号某销售日期电脑销售总额并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务45 学生选课信息管理系统

学生选课的信息包括：编号、课程编号、课程名称、任课教师、学号、姓名、所属院系、学分、成绩等。系统能够完成学生选课信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有学生选课的信息进行排序。

（2） 查询：分别按课程编号、课程名称、任课教师条件查找学生选课信息。

（3） 更新：按编号对某个学生选课的某项信息进行修改。

（4） 插入：加入新学生选课的信息。

（5） 删除：按编号删除已注销的学生选课信息。

（6） 浏览：输出所有学生选课信息。

（7） 统计：统计某学生选修某院系课程的门数并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务46 网上书店管理系统

图书购买的信息包括：购买编号、图书编号、图书种类、图书名称、单价、姓名、性别、年龄、联系方式等。系统能够完成图书购买信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按购买编号对所有图书购买的信息进行排序。

（2） 查询：分别按购买编号、图书编号、图书名称条件查找图书购买信息。

（3） 更新：按购买编号对某个图书购买的某项信息进行修改。

（4） 插入：加入新图书购买的信息。

（5） 删除：按购买编号删除已注销的图书购买信息。

（6） 浏览：输出所有图书购买信息。

（7） 统计：统计某图书种类某图书名称的销售总额并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务47 教室信息管理系统

教室的信息包括：编号、容纳人数、类型、位置、空闲的开始时间、结束时间、教室设备编号、设备状态等。系统能够完成教室信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有教室的信息进行排序。

（2） 查询：分别按编号、容纳人数、类型条件查找教室信息。

（3） 更新：按编号对某个教室的某项信息进行修改。

（4） 插入：加入新教室的信息。

（5） 删除：按编号删除已注销的教室信息。

（6） 浏览：输出所有教室信息。

（7） 统计：统计某类型某容纳人数的教室的数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务48 论坛管理信息系统

贴子信息包括：贴子编号、发贴日期、时间、等级、作者昵称等。系统能够完成贴子信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按贴子编号对所有贴子的信息进行排序。

（2） 查询：分别按贴子编号、发贴日期、作者昵称条件查找贴子信息。

（3） 更新：按贴子编号对某个贴子信息的某项信息进行修改。

（4） 插入：加入新贴子的信息。

（5） 删除：按贴子编号删除不合规的贴子的信息。

（6） 浏览：输出所有贴子信息。

（7） 统计：统计某作者某发贴日期发贴的数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务49 作者信息管理系统

作者信息包括：作者编号、作者昵称、性别、年龄、职业、爱好、发贴数量等。系统能够完成作者信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按作者编号对所有作者信息进行排序。

（2） 查询：分别按作者编号、作者昵称、职业条件查找作者信息。

（3） 更新：按作者编号对某个作者信息的某项信息进行修改。

（4） 插入：加入新作者的信息。

（5） 删除：按作者编号删除已注销的作者信息。

（6） 浏览：输出所有作者信息。

（7） 统计：统计某年龄某职业作者的数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务50 通讯录信息管理系统

通讯录信息包括：编号、通讯人姓名、联系方式、工作地点、城市、备注等。系统能够完成通讯录信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有通讯录的信息进行排序。

（2） 查询：分别按编号、通讯人姓名、城市条件查找通讯录信息。

（3） 更新：按编号对某个通讯录信息的某项信息进行修改。

（4） 插入：加入新通讯录的信息。

（5） 删除：按编号删除已注销的通讯录信息。

（6） 浏览：输出所有通讯录信息。

（7） 统计：统计某城市某工作地点的通讯人的数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务51 备忘录信息管理系统

备忘录信息包括编号、时间、事件、地点、备注等。系统能够完成备忘录信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有备忘录的信息进行排序。

（2） 查询：分别按编号、时间、事件条件查找备忘录信息。

（3） 更新：按编号对某个备忘录的某项信息进行修改。

（4） 插入：加入新备忘录的信息。

（5） 删除：按编号删除已注销备忘录的信息。

（6） 浏览：输出所有备忘录信息。

（7） 统计：统计某地点某事件的备忘录数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务52 个人财物信息管理系统

个人财物信息包括编号、时间、收入项目、收入金额，消费项目、消费金额、剩余资金等。系统能够完成个人财物信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有个人财物的信息进行排序。

（2） 查询：分别按编号、时间、收入项目条件查找个人财物信息。

（3） 更新：按编号对某个个人财物的某项信息进行修改。

（4） 插入：加入新个人财物的信息。

（5） 删除：按编号删除已注销的个人财物信息。

（6） 浏览：输出所有个人财物信息。

（7） 统计：统计某时间段收入总额和消费总额并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务53日记信息管理系统

日记信息包括编号、时间、地点、事情、人物、备注等。系统能够完成日记信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有日记的信息进行排序。

（2） 查询：分别按编号、时间、地点条件查找日记信息。

（3） 更新：按编号对某个日记的某项信息进行修改。

（4） 插入：加入新日记的信息。

（5） 删除：按编号删除已注销的日记信息。

（6） 浏览：输出所有日记信息。

（7） 统计某地点某人物日志信息的数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务54轿车信息管理系统

轿车信息包括：轿车编号、型号、颜色、生产厂家、出厂日期、价格等。系统能够完成轿车信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按轿车编号对所有轿车的信息进行排序。

（2） 查询：分别按轿车编号、型号、生产厂家条件查找轿车信息。

（3） 更新：按轿车编号对某个轿车的某项信息进行修改。

（4） 插入：加入新轿车的信息。

（5） 删除：按轿车编号删除已注销的轿车信息。

（6） 浏览：输出所有轿车信息。

（7） 统计：统计某生产厂家某价格的轿车信息的数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务55客户信息管理系统

客户信息包括：编号、客户名称、联系方式、地址、所在城市、联系事件记录等。系统能够完成客户信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按不同关键字，对所有客户的信息进行排序。

（2） 查询：按编号、客户名称、地址条件查找客户信息。

（3） 更新：按编号对某个客户的某项信息进行修改。

（4） 插入：加入新客户的信息。

（5） 删除：按编号删除已注销的客户信息。

（6） 浏览：输出所有客户信息。

（7） 统计：统计某所在城市某地址的客户信息数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务56轿车销售信息管理系统

轿车销售信息包括：编号、销售日期、轿车类型、颜色、数量、经手人等。系统能够完成轿车销售信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有轿车销售的信息进行排序。

（2） 查询：分别按编号、销售日期、轿车类型条件查找轿车销售信息。

（3） 更新：按编号对某个轿车销售的某项信息进行修改。

（4） 插入：加入新轿车销售的信息。

（5） 删除：按编号删除已注销的轿车销售信息。

（6） 浏览：输出所有轿车销售信息。

（7） 统计：统计某销售日期某经手人销售轿车总数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务57药品信息管理系统

药品信息包括：药品编号、药品名称、生产厂家、生产日期、保质期、用途、价格、数量、经手人等。系统能够完成药品信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按药品编号对所有药品的信息进行排序。

（2） 查询：分别按药品编号、药品名称、生产厂家条件查找药品信息。

（3） 更新：按药品编号对某个药品的某项信息进行修改。

（4） 插入：加入新药品的信息。

（5） 删除：按药品编号删除已注销的药品信息。

（6） 浏览：输出所有药品信息。

（7） 统计：统计某药品名称某生产日期药品总数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务58药品销售信息管理系统

药品销售信息包括：购买编号、药品编号、名称、客户编号、姓名、联系方式、购买时间、数量等。系统能够完成药品销售信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按购买编号对所有药品销售的信息进行排序。

（2） 查询：分别按购买编号、名称、客户编号条件查找药品销售信息。

（3） 更新：按购买编号对某个药品销售的某项信息进行修改。

（4） 插入：加入新药品销售的信息。

（5） 删除：按购买编号删除已注销的药品销售信息。

（6） 浏览：输出所有药品销售信息。

（7） 统计：统计某客户编号某购买时间购买的药品总数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务59文件信息管理系统

文件信息包括：文件编号、文件种类、文件名称、存放位置、存放日期、保管人等。系统能够完成文件信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按文件编号对所有文件的信息进行排序。

（2） 查询：分别按文件编号、文件种类、文件名称条件查找文件信息。

（3） 更新：按文件编号对某个文件的某项信息进行修改。

（4） 插入：加入新文件的信息。

（5） 删除：按文件编号删除已注销的文件信息。

（6） 浏览：输出所有文件信息。

（7） 统计：统计某文件种类某存放日期的文件数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务60会议记录信息管理系统

会议记录信息包括：编号、会议时间、参会人、记录员、会议地点、会议内容等。系统能够完成会议记录信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有会议记录的信息进行排序。

（2） 查询：分别按编号、会议时间、会议地点条件查找会议记录信息。

（3） 更新：按编号对某个会议记录的某项信息进行修改。

（4） 插入：加入新会议记录的信息。

（5） 删除：按编号删除已注销的会议记录信息。

（6） 浏览：输出所有会议记录信息。

（7） 统计：统计某会议地点某会议内容的会议记录数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务61 办公室日常事务信息管理系统

办公室日常事务信息包括：编号、时间、事务、记录人、备注等。系统能够完成办公室日常事务信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有办公室日常事务的信息进行排序。

（2） 查询：分别按编号、时间、事务条件查找办公室日常事务信息。

（3） 更新：按编号对某个办公室日常事务的某项信息进行修改。

（4） 插入：加入新办公室日常事务的信息。

（5） 删除：按编号删除已注销的办公室日常事务信息。

# （6） 浏览：输出所有办公室日常事务信息。

（7） 统计：统计某段时间内日常事务信息的数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务62 超市会员信息管理系统

超市会员的信息包括：编号、姓名、性别、年龄、工作单位、联系方式、等级、积分等。系统能够完成会员信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有会员的信息进行排序。

（2） 查询：分别按编号、姓名、年龄条件查找会员信息。

（3） 更新：按编号对某个会员的某项信息进行修改。

（4） 插入：加入新会员的信息。

（5） 删除：按编号删除已注销的会员信息。

（6） 浏览：输出所有会员信息。

（7） 统计：统计某性别某等级的超市会员的数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务63 超市会员购物信息管理系统

超市会员购物的信息包括：购买编号、物品编号、物品名称、所属种类、数量、价格、会员编号、姓名、返利信息等。系统能够完成会员购物信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按购买编号对所有会员购物的信息进行排序。

（2） 查询：统计按购买编号、物品编号、物品名称条件查找会员购物信息。

（3） 更新：按购买编号对某个会员购物的某项信息进行修改。

（4） 插入：加入新会员购物的信息。

（5） 删除：按购买编号删除已注销的会员购物信息。

（6） 浏览：输出所有会员购物信息。

（7） 统计：统计某会员购买某物品的总数量和总金额并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务64 高校科研信息管理系统

高校科研的信息包括：教师编号、姓名、性别、学历、所属部门、研究方向、课题研究情况、专利、论文及著作发表情况等。系统能够完成科研信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按教师编号对所有科研的信息进行排序。

（2） 查询：分别按教师编号、姓名、所属部门条件查找科研信息。

（3） 更新：按教师编号对某个科研的某项信息进行修改。

（4） 插入：加入新的科研信息。

（5） 删除：按教师编号删除已注销的科研信息。

（6） 浏览：输出所有科研信息。

（7） 统计：统计某所属部门某研究方向的科研信息数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务65 长途汽车信息管理系统

长途汽车的信息包括：编号、出发地、目的地、出发时间、所需时间、汽车种类、票价、最大载客量等。系统能够完成长途汽车信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有长途汽车的信息进行排序。

（2） 查询：分别按编号、出发地、目的地条件查找长途汽车信息。

（3） 更新：按编号对某个长途汽车的某项信息进行修改。

（4） 插入：加入新长途汽车的信息。

（5） 删除：按编号删除长途汽车的信息。

（6） 浏览：输出所有长途汽车信息。

（7） 统计：统计某出发地某目的地的长途汽车数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务66 机票预定信息管理系统

机票预定的信息包括：流水号、航班编号、飞机名称、机舱等级、票价、折扣、经手业务员、客户姓名、联系方式、证件号码等。系统能够完成员工信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按流水号对所有机票预定的信息进行排序。

（2） 查询：分别按航班编号、飞机名称、机舱等级条件查找机票预定信息。

（3） 更新：按流水号对某个机票预定的某项信息进行修改。

（4） 插入：加入新机票预定的信息。

（5） 删除：按流水号删除已注销的机票预定信息。

（6） 浏览：输出所有机票预定信息。

（7） 统计：统计某客户乘坐某类型飞机的次数并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务67 教师业绩（工作量）信息管理系统

教师业绩的信息包括：编号、教师工号、姓名、性别、学院、名称、类别、备注、工作量等。系统能够完成教师业绩信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有教师业绩的信息进行排序。

（2） 查询：分别按教师工号、姓名、学院条件查找教师业绩信息。

（3） 更新：按编号对某个教师业绩的某项信息进行修改。

（4） 插入：加入新教师业绩的信息。

（5） 删除：按编号删除已离职的教师业绩的信息。

（6） 浏览：输出所有教师业绩信息。

（7） 统计：统计某学院某姓名的教师总工作量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务68 学生公寓信息管理系统

学生公寓的信息包括：编号、类型、位置、面积、床位数、人数、空闲床位数等。系统能够完成学生公寓信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有学生公寓的信息进行排序。

（2） 查询：分别按编号、类型、位置条件查找学生公寓信息。

（3） 更新：按编号对某个学生公寓的某项信息进行修改。

（4） 插入：加入新学生公寓的信息。

（5） 删除：按编号删除已注销的学生公寓信息。

（6） 浏览：输出所有学生公寓信息。

（7） 统计：统计某类型某位置公寓总床位数和总空闲床位数并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务69 邮局订报信息管理系统

邮局订报的信息包括：编号、报纸编号、报纸名称、报纸单价、客户姓名、客户电话、客户地址、订购份数、开始订购时间、订购截止等。系统能够完成邮局订报信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有邮局订报的信息进行排序。

（2） 查询：分别按编号、报纸编号、报纸名称条件查找邮局订报信息。

（3） 更新：按编号对某个邮局订报的某项信息进行修改。

（4） 插入：加入新邮局订报的信息。

（5） 删除：按编号删除已注销的邮局订报信息。

（6） 浏览：输出所有邮局订报信息。

（7） 统计：统计某客户订购某出版社报纸总份数并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务70 光盘出租信息管理系统

光盘出租的信息包括：编号、光盘编号、光盘名字、出版社、类别、借出数量、起租日期、归还日期、租赁人、归还状态等。系统能够完成光盘出租信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有光盘出租的信息进行排序。

（2） 查询：分别按编号、光盘编号、光盘名字条件查找光盘出租信息。

（3） 更新：按编号对某个光盘出租的某项信息进行修改。

（4） 插入：加入新光盘出租的信息。

（5） 删除：按编号删除已注销的光盘出租信息。

（6） 浏览：输出所有光盘出租信息。

（7） 统计：统计某租赁人租赁某出版社的光盘数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务71 停车场信息管理系统

停车场的信息包括：编号、车牌号、类型、车位编号、使用日期、入场时间、出场时间、单价、费用等。系统能够完成停车场信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有停车场的信息进行排序。

（2） 查询：分别按编号、车牌号、类型条件查找停车场信息。

（3） 更新：按编号对某个停车场信息的某项信息进行修改。

（4） 插入：加入新停车场的信息。

（5） 删除：按编号删除已注销的停车场信息。

（6） 浏览：输出所有停车场信息。

（7） 统计：统计某车位编号某使用日期的总费用并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务72 旅行社资源信息管理系统

旅行社资源的信息包括：编号、名称、旅行社介绍、等级、联系电话、地址、联系人、主要旅游路线等。系统能够完成旅行社资源信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有旅行社资源的信息进行排序。

（2） 查询：分别按编号、名称、等级条件查找旅行社资源信息。

（3） 更新：按编号对某个旅行社资源的某项信息进行修改。

（4） 插入：加入新旅行社资源的信息。

（5） 删除：按编号删除已注销的旅行社资源的信息。

（6） 浏览：输出所有旅行社资源信息。

（7） 统计：统计某等级某地址的旅行社资源信息的数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务73 旅游城市信息管理系统

旅游城市的信息包括：编号、城市名称、所在省、所在地区、城市人文、城市历史、城市地理、主要旅游路线、主要景点等。系统能够完成旅游城市信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有旅游城市的信息进行排序。

（2） 查询：分别按编号、城市名称，城市人文条件查找旅游城市信息。

（3） 更新：按编号对某个旅游城市的某项信息进行修改。

（4） 插入：加入新旅游城市的信息。

（5） 删除：按编号删除已注销的旅游城市信息。

（6） 浏览：输出所有旅游城市信息。

（7） 统计：统计某省某地区旅游城市的数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务74 旅游景点信息管理系统

旅游景点的信息包括：编号、景点名称、景点来源、景点历史、景点图片、地点、所在地区、门票价格、优惠条件等。系统能够完成旅游景点信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有旅游景点的信息进行排序。

（2） 查询：分别按编号、景点名称、地点条件查找旅游景点信息。

（3） 更新：按编号对某个旅游景点的某项信息进行修改。

（4） 插入：加入新旅游景点的信息。

（5） 删除：按编号删除已注销的旅游景点的信息。

（6） 浏览：输出所有旅游景点信息。

（7） 统计：统计某地区某地点旅游景点的数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务75 餐饮服务信息管理系统

餐饮的信息包括：编号、餐桌号、菜单、就餐情况、点餐时间、金额等。系统能够完成餐饮信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有餐饮的信息进行排序。

（2） 查询：分别按编号、餐桌号、菜单条件查找餐饮信息。

（3） 更新：按编号对某个餐饮的某项信息进行修改。

（4） 插入：加入新餐饮的信息。

（5） 删除：按编号删除已注销的餐饮的信息。

（6） 浏览：输出所有餐饮信息。

（7） 统计：统计某餐桌号某菜单的总金额并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务76 小区业主信息管理系统

业主信息的信息包括：业主ID、业主姓名、出生年月、性别、工作单位、身份证号码、联系电话、入住时间、迁出时间等。系统能够完成业主信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按业主ID对所有业主的信息进行排序。

（2） 查询：分别按业主ID、业主姓名、出生年月条件查找业主信息。

（3） 更新：按业主ID对某个业主的某项信息进行修改。

（4） 插入：加入新业主的信息。

（5） 删除：按业主ID删除已注销的业主的信息。

（6） 浏览：输出所有业主信息。

（7） 统计：统计某工作单位某入住时间的业主数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务77 小区房产信息管理系统

小区房产的信息包括：房间编号、物业地址、建筑面积、使用面积、总价、是否己经售出、房主编号、房主姓名等。系统能够完成小区房产信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按房间编号对所有小区房产的信息进行排序。

（2） 查询：分别按房间编号、房主姓名、建筑面积条件查找小区房产信息。

（3） 更新：按房间编号对某个小区房产的某项信息进行修改。

（4） 插入：加入新小区房产的信息。

（5） 删除：按房间编号删除已注销的小区房产的信息。

（6） 浏览：输出所有小区房产信息。

（7） 统计：统计某房主姓名某建筑面积房屋的数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务78 小区物业设备信息管理系统

物业设备的信息包括：设备编号、设备名称、型号、品牌、所属部门、购买日期、事故记录等。系统能够完成物业设备信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按设备编号对所有物业设备的信息进行排序。

（2） 查询：分别按设备编号、设备名称、所属部门条件查找物业设备信息。

（3） 更新：按设备编号对某个物业设备的某项信息进行修改。

（4） 插入：加入新物业设备的信息。

（5） 删除：按设备编号删除已注销的物业设备的信息。

（6） 浏览：输出所有物业设备信息。

（7） 统计：统计某型号某品牌的设备数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务79 小区物业仪表收费管理系统

小区物业仪表收费的信息包括：收费编号、仪表编号、住户姓名、年份、月份、用量、单价、费用、抄表日期、交费日期、办理人等。系统能够完成仪表收费信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按收费编号对所有仪表收费的信息进行排序。

（2） 查询：分别按收费编号、仪表编号、住户姓名条件查找仪表收费信息。

（3） 更新：按收费编号对某个仪表收费的某项信息进行修改。

（4） 插入：加入新仪表收费的信息。

（5） 删除：按收费编号删除已注销的仪表收费的信息。

（6） 浏览：输出所有仪表收费信息。

（7） 统计：统计某年份某月份的总费用并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务80 小区用户报修管理系统

用户报修的信息包括：报修编号、报修日期、接待人、报修住户、电话、完成日期、维修人员、维修费用、报修内容、维修情况等。系统能够完成用户报修信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按报修编号对所有用户报修的信息进行排序。

（2） 查询：分别按报修编号、报修日期、报修住户条件查找用户报修信息。

（3） 更新：按报修编号对某个用户报修的某项信息进行修改。

（4） 插入：加入新用户报修的信息。

（5） 删除：按报修编号删除已注销的用户报修的信息。

（6） 浏览：输出所有用户报修信息。

（7） 统计：统计某维修人员完成的某报修日期房屋维护总费用。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务81 小区住户投诉管理系统

住户投诉的信息包括：投诉编号、投诉日期、接待人、投诉住户、物业地址、电话、处理日期、处理人员、投诉内容、处理情况等。系统能够完成住户投诉信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按投诉编号对所有住户投诉的信息进行排序。

（2） 查询：分别按投诉编号、投诉日期投诉住户条件查找住户投诉信息。

（3） 更新：按投诉编号对某个住户投诉的某项信息进行修改。

（4） 插入：加入新住户投诉的信息。

（5） 删除：按投诉编号删除已注销的住户投诉的信息。

（6） 浏览：输出所有住户投诉信息。

（7） 统计：统计某投诉日期某投诉内容的投诉信息数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务82 小区物业收费维修管理系统

物业收费的信息包括：收费编号、住户姓名、年份、月份、收费项目、收费方式、应收金额、已交金额、欠款金额、交费日期、办理人等。系统能够完成物业收费信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按收费编号对所有物业收费的信息进行排序。

（2） 查询：分别按收费编号、住户姓名、收费项目特定条件查找物业收费信息。

（3） 更新：按收费编号对某个物业收费的某项信息进行修改。

（4） 插入：加入新物业收费的信息。

（5） 删除：按收费编号删除已注销的物业收费的信息。

（6） 浏览：输出所有物业收费信息。

（7） 统计：统计某年份某月份费用应收总额和已交总额并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务83 小区物业设备维修管理系统

物业设备维修的信息包括：维修编号、设备编号、设备名称、维修日期、完成日期、维修费用、维修人员、维修内容、维修情况等。系统能够完成物业设备维修信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按维修编号对所有物业设备维修的信息进行排序。

（2） 查询：分别按维修编号、设备名称、维修日期条件查找物业设备维修信息。

（3） 更新：按维修编号对某个物业设备维修的某项信息进行修改。

（4） 插入：加入新物业设备维修的信息。

（5） 删除：按维修编号删除已注销的物业设备维修的信息。

（6） 浏览：输出所有物业设备维修信息。

（7） 统计：统计某设备名称某完成日期的维修费用总额并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务84 学校固定资产信息管理系统

固定资产信息包括：编号、资产名称、资产分类、所属部门、购置方式、金额、使用人、存放地点等。系统能够完成固定资产信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有固定资产信息进行排序。

（2） 查询：分别按编号、资产名称、使用人条件查找固定资产信息。

（3） 更新：按编号对某个固定资产的某项信息进行修改。

（4） 插入：加入新固定资产的信息。

（5） 删除：按编号删除已注销的固定资产的信息。

（6） 浏览：输出所有固定资产信息。

（7） 统计：统计某使用人某类资产的数量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务85 职工工资信息管理系统

职工工资的信息包括：工资编号、职工号、姓名、部门、学历、职务、年月、基本工资、绩效工资、扣税、扣水电、实发工资等。系统能够完成职工工资信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按工资编号对所有职工工资的信息进行排序。

（2） 查询：分别按工资编号、姓名、部门条件查找职工工资信息。

（3） 更新：按工资编号对某个职工工资的某项信息进行修改。

（4） 插入：加入新职工工资的信息。

（5） 删除：按工资编号删除已离职的职工工资的信息。

（6） 浏览：输出所有职工工资信息。

（7） 统计：统计某部门某年月实发工资总额并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务86 高校教材订购信息管理系统

高校教材订购的信息包括：订购编号、学年、学期、课程名称、教材名称、书号（ISBN）、出版社、出版时间、作者、订购数量、教材使用班级等。系统能够完成教材订购信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按订购编号对所有教材订购的信息进行排序。

（2） 查询：分别按订购编号、学年、学期条件查找教材订购信息。

（3） 更新：按订购编号对某个教材订购的某项信息进行修改。

（4） 插入：加入新教材订购的信息。

（5） 删除：按订购编号删除已注销的教材订购信息。

（6） 浏览：输出所有教材订购信息。

（7） 统计：统计某学年某学期订购教材总量并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务87 某自来水公司收费管理系统

自来水公司收费的信息包括：收费编号、客户、月份、用水类别、用水量、水价、应收费用、实收费用、业务员等。系统能够完成自来水公司收费信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按收费编号对所有自来水公司收费的信息进行排序。

（2） 查询：分别按收费编号、客户、月份条件查找自来水公司收费信息。

（3） 更新：按收费编号对某个自来水公司收费的某项信息进行修改。

（4） 插入：加入新自来水公司收费的信息。

（5） 删除：按收费编号删除已注销的自来水公司收费的信息。

（6） 浏览：输出所有自来水公司收费信息。

（7） 统计：统计某客户某月份总用水量和实收总费用并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务88 某电力公司收费管理信息系统

电力公司收费的信息包括：收费编号、客户、月份、用电类别、用电度数、电价、应收费用、实收费用、业务员等。系统能够完成电力公司收费信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按收费编号对所有电力公司收费的信息进行排序。

（2） 查询：分别按收费编号、客户、月份条件查找电力公司收费信息。

（3） 更新：按收费编号对某个电力公司收费的某项信息进行修改。

（4） 插入：加入新电力公司收费的信息。

（5） 删除：按收费编号删除已注销的电力公司收费的信息。

（6） 浏览：输出所有电力公司收费信息。

（7） 统计：统计某客户某月份用电总度数和实收总费用并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务89 职业介绍信息管理系统

职业介绍的信息包括：编号、职业类型号、用人单位、需求人数、已聘人数、备注等。系统能够完成职业介绍信息的查询、更新、插入、删除、排序等功能。

（1） 排序：按编号对所有职业介绍的信息进行排序。

（2） 查询：分别按编号、职业类型号、用人单位条件查找职业介绍信息。

（3） 更新：按编号对某个职业介绍的某项信息进行修改。

（4） 插入：加入新职业介绍的信息。

（5） 删除：按编号删除已注销的职业介绍信息。

（6） 浏览：输出所有职业介绍信息。

（7） 统计：统计某职业类型号某用人单位的需求人数总数和已聘人数总数并输出。

（8）保存：数据永久性存放至外存。

（9）读取：将存放于外存的数据读入内存继续操作。

# 任务90 单项选择题标准化考试系统

为单项选择题标准化考试设计程序，每个试题包括试题编号、题干、4个备选答案、标准答案，以文件保存试题库，能实现以下功能。

（1）试题录入：可随时增加试题到试题库中。

（2）试题抽取：每次从试题库中可以随机抽出N道题（N由键盘输入）。

（3）答题：用户可实现输入自己的答案。

（4）自动判卷：系统可根据用户答案与标准答案的对比实现判卷并给出成绩。

（5）试题浏览：输出所有试题信息。

（6）试题查询：按试题编号查询试题信息。

（7）试题维护：按试题编号删除和修改试题信息。

# 任务91 各种排序算法时间性能的比较

**1) 问题描述**

对各种排序方法（直接插入排序、希尔排序、起泡排序、快速排序、直接选择排序、堆排序和归并排序）的时间性能进行比较。

**2) 基本要求**

(1) 设计并实现上述各种排序算法；

(2) 产生正序和逆序的初始排列分别调用上述排序算法，并比较时间性能；

(3) 产生随机的初始排列分别调用上述排序算法，并比较时间性能。

上述各种排序方法都是基于比较的内排序，其时间主要消耗在排序过程中进行的记录的比较次数和移动次数，因此，统计在相同数据状态下不同排序算法的比较次数和移动次数，即可实现比较各种排序算法的目的。

**【思考题】**如果测算每种排序算法所用实际的时间，应如何修改排序算法？

# 任务92 二叉排序树的遍历（ 用递归或非递归的方法都可以）

**1)问题描述**

输入树的各个结点，建立二叉排序树，对建立的二叉排序树进行查找、层次、先序、中序和后序遍历并统计该二叉树中叶子结点的数目。

**2)基本要求**

(1)用菜单实现

(2)能够输入树的各个结点，并能够输出用不同方法遍历的遍历序列和叶子结点的数目。

# 任务93 模拟学校招生的程序设计（专业优先）

我校有10个专业招生，每个专业都有计划招生数，达到我校录取分数线的学生，成绩及志愿花名册已由市考试院送达我校（按计划人数的1.1倍投档），每个学生均报3个志愿。

录取原则：按专业优先的原则录取，按成绩由高到低先录取第一志愿的专业，若考生第一志愿未被录取，在第二志愿专业未录满的情况下则按成绩由高到低录取第二志愿专业，以此类推。若学生3个志愿都未被录取，则看是否同意调剂，调剂专业方式可以根据分数高的学生调剂到招生人数越接近计划数的专业。

程序最终输出录取与未录取的结果（原数据存于文件中，录取结果存入文件中）。

# 任务94模拟学校招生的程序设计（分数优先）

我校有10个专业招生，每个专业都有计划招生数，达到我校录取分数线的学生，成绩及志愿花名册已由市考试院送达我校（按计划人数的1.1倍投档），每个学生均报3个志愿。

录取原则：按分数优先的原则录取，按考分成绩由高到低排序依次录取，若第一志愿的专业已录满，则该考生的分数减5分重新排序，录取第二志愿专业，以此类推。若学生3个志愿都未被录取，则看是否同意调剂，调剂专业方式可以根据分数（原考分）高的学生调剂到招生人数越接近计划数的专业。

程序最终输出录取与未录取的结果（原数据存于文件中，录取结果存入文件中）。

# 任务95 大整数的链式存储及实现

采用链接方式储存一个大的正整数，如：

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1234 |  |  | 5678 |  |  | 9011 |  |  | 1213 | ∧ |

表示正整数： 1213901156781234

设计一个实现完成两个任意长正整数的加、减、乘、除运算的程序。在这里长整数没有范围限制，可任意长。运算之后的进位、借位等都要进行正确处理，可实现动态的输入，实时的输出。

要求初始有文本菜单可供选择加、减、乘、除的功能。

# 任务96 评奖系统设计

为教务处设计一个学生评价老师的程序：

每位学生投一张票，选出自己最喜爱的老师，选票格式为：

|  |  |  |
| --- | --- | --- |
| 第一喜爱的老师 | 第二喜爱的老师 | 第三喜爱的老师 |
| 工号 | 工号 | 工号 |

要有学生投票信息的录入、浏览、查询（按学号、按姓名）、修改（按学号）、删除（按学号）和排序（按学号）功能，上述数据存放在一个数据文件中。

根据票中情况给老师记分，记分规则为：

第一喜爱的老师：8分

第二喜爱的老师：5分

第三喜爱的老师：3分

请统计各位老师的总分，并由高到低顺序输出到一个文件中，如得分相同时，则优先考虑第一喜爱的老师，依次类推。

设老师数<=15， 工号分别是1，2，3……..15

学生数<=100

# 任务97 文件的加密和解密（利用顺序栈）

运行文件加密程序，输入要加密的文件名，然后输入密码，最后输入加密后的文件名，程序对文件中读入的每一个字符与密码进行异或，再将异或后的内容倒序写入指定的文件中。

解密程序为加密程序的逆过程。

请使用顺序栈的结构设计并完成程序的功能。

注：此题目由1人独立完成。

# 任务98 文件的加密和解密（利用链栈）

运行文件加密程序，输入要加密的文件名，然后输入密码，最后输入加密后的文件名，程序对文件中读入的每一个字符与密码进行异或，再将异或后的内容倒序写入指定的文件中。

解密程序为加密程序的逆过程。

请使用链栈的结构设计并完成程序的功能。

注：此题目由1人独立完成。

# 任务99 常量算术表达式的计算（利用顺序栈）

编写一个程序，对于输入的一个常量算术表达式的字符串，（运算符只有双目运算符+、-、\*、/、%，运算数只为int，字符串以“=“结尾），计算该表达式的值。

注：此题目由1人独立完成。

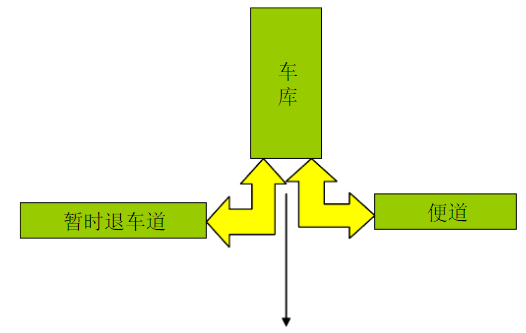
# 任务100 停车场管理

设停车场内只有一个可停放n辆汽车的狭长通道，且只有一个大门可供汽车进出。汽车在停车场内按车辆到达时间的先后顺序，依次由北向南排列（大门在最南端，最先到达的第一辆车停放在车场的最北端），若车场内已停满n辆汽车，则后来的汽车只能在门外的便道上等候，一旦有车开走，则排在便道上的第一辆车即可开入；当停车场内某辆车要离开时，在它之后开入的车辆必须先退出车场为它让路，待该辆车开出大门外，其它车辆再按原次序进入车场，每辆停放在车场的车在它离开停车场时必须按它停留的时间长短交纳费用。试为停车场编制按上述要求进行管理的模拟程序。

测试数据：设n=2,输入数据为：（‘A’，1，5），（‘A’，2，10），（‘D’，1，15），（‘A’，3， 20）， （‘A’，4，25），（‘A’，5，30），（‘D’，2，35），（‘D’，4，40），（‘E’，0，0）。每一组输入数据包括三个数据项：汽车“到达”或“离去”信息、汽车牌照号码及到达或离去的时刻，其中，‘A’表示到达；‘D’表示离去，‘E’表示输入结束。

[实现提示]需另设一个栈，临时停放为给要离去的汽车让路而从停车场退出来的汽车，也可用顺序存储结构实现。输入数据按到达或离去的时刻有序。栈中每个元素表示一辆汽车，包含两个数据项：汽车的牌照号码和进入停车场的时刻。

注：此题目由1人独立完成。



# 任务101舞伴搭配问题

班级里有m个女生,有n个男生(m不等于n),现要开一个舞会。 男女生分别编号坐在舞池的两边的椅子上。每曲开始时,依次从男生和女生中各出p人配对跳舞(p≥1), 本曲没成功配对者坐着等待下一曲找舞伴。请使用循环队列，设计程序模拟上述过程，要求如下:

（1）输出每曲配对情况

（2）计算出任何一个男生(编号为X)和任意女生(编号为Y),在第K曲配对跳舞的情况，至少求出K的两个值。

注：此题目由1人独立完成。

# 任务102 修理牧场

农夫John要修理牧场的一段栅栏，他测量了栅栏，发现需要N块木头，每块木头长度为整数Li个单位。于是他购买了一条很长的，能锯成N块的木头（即该木头长度是Li的总和）。农夫John没有锯子，于是他去找农夫Don借锯子。Don是一个守财奴，他要收费，锯一段木头要支付的费用等于这段木头的长度，即锯长度为21的木头就要支付21美分。（例如，要将长度为21的木头锯成长度为8，5，8的三段。第一次锯木头，将木头锯成13和8，花费21；第二次将长度为13的那块木头锯成8和5，花费13，这样总的花费为21+13=34。）但是，如果将长度为21的木头第一次锯成16和5，第二次锯长度为16的木头，总的花费为21+16=37（大于34）。请你帮助John确定锯N块木头所要花费的最少的钱。

[实现提示] 由于木块锯一次产生两块木块，因此锯木过程可以用一棵二叉树表示：根表示初始木板，初始木板的总长度为根结点的权，n段目标木板为n个叶结点，其中第i个叶结点的权为第i段目标木板的长度wi。根据题意，计算总花费最小的锯木方案，实际上是计算带权路径长度和最小的哈夫曼树。

注：此题目由1人独立完成。

# 任务103 C源程序关键字统计及Huffman编码设计

对于任意一个c语言的源程序，统计其关键词（c语言专用的标识符）出现的次数，并以该次数作为权重，求出其对应的Huffman编码。

（1）设标识符在原文件中前后都有空格字符隔开。

（2）更通常情况，只要求符合c语言语法即可。

注：此题目由1人独立完成。

# 任务104 家谱处理

人类学研究对于家族很感兴趣，于是研究人员搜集了一些家族的家谱进行研究。实验中，研究人员将家谱转换为文本文件。下面为家谱文本文件的实例：

John

Robert

Frank

Andrew

Nancy

David

家谱文本文件中，每一行包含一个人的名字。第一行中的名字是这个家族最早的祖先。家谱仅包含最早祖先的后代，而他们的丈夫或妻子不出现在家谱中。每个人的子女比父母多缩进2个空格。以上述家谱文本文件为例，John是这个家族最早的祖先，他有两个子女Robert和Nancy，Robert有两个子女Frank和Andrew，Nancy只有一个子女David。

在实验中，研究人员还收集了家庭文件，并提取了家谱中有关两个人关系的陈述语句。下面为家谱中关系的陈述语句实例：

John is the parent of Robert

Robert is a sibling of Nancy

David is a descendant of Robert

研究人员需要判断每个陈述语句是真还是假，请编写程序帮助研究人员判断。

注：此题目由1人独立完成。

# 任务105 哈希表设计

为周围熟悉的30个人名设计一个哈希表，假设人名为姓名的汉语拼音形式，哈希函数用除留余数法构造，用线性探测再散列法或链地址法处理冲突，完成相应的建表和查表程序，使得平均查找长度不超过 2。

选作内容

（1） 从教科书上介绍的集中哈希函数构造方法中选出适用者并设计几个不同的哈希函数，比较他们的地址冲突率（可以用更大的名字集合作实验）。

（2） 研究这30个人名的特点，努力找一个哈希函数，使得对于不同的拼音名一定不发生地址冲突。

（3） 在哈希函数确定的前提下尝试各种不同处理冲突的方法，考察平均查找长度的变化。

注：此题目由1-2人完成。

# 任务106 隔离的共和国

在garlebi海中有很多个香蕉共和国（方便起见，将它们标号为1,2,...,N），gabriel就生活在其中最可爱的苹果焦共和国（标号为M）中~这些香蕉共和国组成了香蕉联盟。由于每个香蕉共和国都是一个孤立的小岛，香蕉共和国之间相互隔离，香蕉联盟计划在香蕉共和国之间建立一些桥梁，以便于香蕉的配种与交易。若两个共和国之间建立了一座桥梁则可从桥梁的一端到达另一端。现在gabriel希望找出，在建立了这些桥梁之后，还有哪些香蕉共和国是与苹果焦共和国相互隔离的。

输入:第一行为两个数N和M，第二行为一个数K表示香蕉联盟计划建立桥梁的数目,接下来的K行每行有两个数A, B(1<=A,B<=N)表示香蕉联盟计划在A，B间建立一座桥梁。

输出:第一行输出一个正整数t表示与苹果蕉共和国隔离的共和国的数量。第二行输出t个递增的正整数表示所有与苹果蕉公共国隔离的共和国，两个正整数间用空格隔开。

**样例:**

**输入:**

3 1

1

1 2

**输出**

1

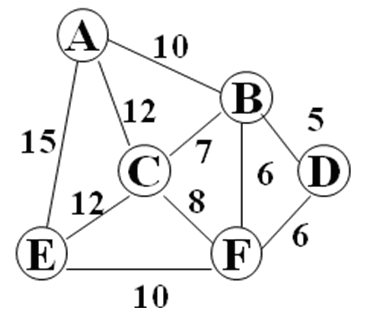
3

注：此题目由1人独立完成。

# 任务107城市之间的通讯网络构建问题

要在 n 个城市间建立通讯网，如何在保证 n 个城市通讯网连通的前题下最节省经费的网络布线方案设计，结果存于数据文件中。

[说明] 以下是六个城市的可以进行布线的无向网络图，该图中边上的数值是两个城市间布线需要的费用，以下图作为测试用例，相关数据放在磁盘文件graph.txt中，布线方案结果存于tree.txt中（要有布线的两个城市名称及费用值）



# 任务108 中心对称问题（利用单链表）

设单链表的表头指针为h，结点结构由data和next两个域构成，其中data域为字符型。写出算法dc(h,n),判断该链表的前n个字符是否中心对称。

[输入] 第1行为n，第二行为字符串。

[输出] 对称输出Yes，否则输出No。

**样例:**

**输入:**

3

2323443

**输出**

Yes

# 任务109 约瑟夫环

约瑟夫（Joeph）问题的一种描述是：编号为1,2,…n的n个人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，从他在顺时针方向上的下一个人开始重新从1报数，如此下去，直至所有人全部出列为止。试设计一个程序求出出列顺序，利用单向循环链表存储结构模拟此过程，按照出列的顺序印出各人的编号。

[测试数据] m的初值为20：密码3,1,7,2,4,8,4

[输出] 6,1,4,7,2,3,5

# 任务110 集合运算

（1）用链表表示两个集合（建立集合链表）。

（2）在集合中增加一个元素。

（3）在集合中删除一个元素。

（4）根据输入元素判断该元素是否属于集合。

（5）对两个集合分别从小到大排序

（6）两个集合合并成另一个新集合，如数值相同，合并为一个数据项

（7）求出两个集合的交集建立一个新的集合。

（8）求出两个集合的差集建立一个新的集合。

**附录2 程序设计实践报告书写格式及内容要求**

**组**

**号**



****

信息技术学院

《程序设计实践》报告

课题名称： 学生考评信息管理系统

班 级 物联网B23-2

指导教师： 陈佳敏

完成日期： 2024年7月7日

|  |  |  |  |
| --- | --- | --- | --- |
|  | 学号 | 姓名 | 成绩 |
| 组长 | 2324096 | 钱信宇 |  |
| 组员1 | 2324103 | 张佳豪 |  |
| 组员2 |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **一、实践目的与要求**  **1、目的**  通过此次实践环节主要达到以下目的：  （1）进一步理解和运用结构化程序设计的思想和方法，学会根据具体问题选择合理的计算机存储结构实现数据的存储，构造较有效率的算法；  （2）学会算法描述的方法，并编制具有结构清晰、合理和易读性的小型实用程序；  （3）并会设计测试方案，完成程序的测试，能撰写出该程序的技术报告，为文档整理工作打下一个初步的基础；  （4）培养小组成员间互相学习，取长补短，协同工作的能力。  **2、要求**  （1）模块化程序设计，锯齿型书写格式，代码要有足够的注释；  （2）根据课题中规定的要求实现既定目标；  （3）撰写格式排版规范、结构完整的技术文档。  **二、课题任务**  （题目、内容及主要功能描述）  **任务 16 学生考评信息管理系统**  **学生考评的信息包括：学号、姓名、性别、出生日期、专业、班级、所属学院、德育考**  **评分、体育考评分、智育考评分、综合考评分等，综合分=0.7\*智育分+0.2\*德育分+0.1\*体**  **育分。系統能够完成学生考评信息的查询、更新、插入、删除、排序等功能。**  **（1）排序：按学号对所有学生考评的信息进行排序。**  **（2）查询：分别按学号、姓名、专业查找学生考评信息。**  **（3）更新，按学号对某个学生考评的某项信息进行修改。**  **（4）插入：加入新学生考评的信息。**  **（5）删除：按学号删除已毕业学生的考评信息。**  **（6）浏览：输出所有学生考评信息。。**  **（7）统计：统计某学院的某专业综合分不及格的学生人数并输出。**  **（8）保存：数据永久性存放至外存。**  **（9）读取：将存放于外存的数据读入内存继续操作。**  **三、总体设计**  **1、存储结构及数据类型定义**  （采用了什么存储结构、被处理数据定义的数据类型，数据类型定义中要有注释说明）  // 定义学生结构体 typedef struct {  int studentID;// 学号  char name[50]; // 姓名，最多包含50个字符  char gender[10];// 性别，最多包含10个字符  char birthDate[20];// 出生日期，最多包含20个字符  char major[50];// 专业，最多包含50个字符  char className[50]; // 班级，最多包含50个字符  char college[50]; // 所属学院，最多包含50个字符  float moralEducationScore; // 德育分，小数类型，用于记录学生的德育成绩  float physicalEducationScore;// 体育分，小数类型，用于记录学生的体育成绩  float intellectualEducationScore;// 智育分，小数类型，用于记录学生的智育成绩  通过calculateComprehensiveScore函数计算得出 comprehensiveScore; }student// 计算综合分数  //定义了一个链表头指针head用于存储学生结构体的链表：Student\*head=NULL  typedef struct node{  student data;  struct student\* next;  }newstudent;  **2、程序结构**  （程序整体的模块结构图）    **3、所实现的功能函数**  （每个功能对应的函数名、函数形参及返回值说明）  calculateComprehensiveScore(Student\*s)  功能：计算学生的综合分数。  形参：Student\*s，表示一个学生的结构体。  返回值：float，表示计算出的综合分数。  addStudent()  功能：添加一个新的学生记录到学生数组中。  deleteStudentByID()  功能：根据学生学号删除学生记录。  updateStudentInfoByID()  功能：根据学号更新学生的德育分、体育分和智育分。  queryStudentInfo()  功能：查询学生的信息。可以选择按学号、姓名或班级查询。  sortStudentsByID()  功能：按照学号对学生记录进行排序。  browseAllStudents()  功能：浏览所有学生的信息。  countFailingStudentsInCollegeAndMajor()  功能：统计某学院某专业中不及格学生的人数。  saveDataToFile()  功能：将学生数据保存到文件中。  loadDataFromFile()  功能：从文件中读取学生数据。  **四、小组成员及分工**  （组长、组员及每人承担的具体模块任务或其他）  **组长钱信宇负责：更新，查询，浏览，统计**  **组员张佳豪负责：添加，删除，排序，保存，读取**  **五、 测试**  （整合各功能模块后的测试结果截图及说明）  1.添加学生    2.删除学生    3.更新学生信息    4.查询学生信息    5.排序学生信息    6.浏览所有学生信息    7.统计不合格学生人数    8.保存数据    9.读取数据 | | | | | | | | | | | |
| **六、个人详细设计及实施、实践总结** | | | | | | | | | | | |
| **学号** | | | | **2324096** | | | **姓名** | | | **钱信宇** | |
| **详**  **细**  **设**  **计**  **及**  **实**  **施** | | **1、算法设计**  （对本人承担的各模块用传统流程图或自然语言描述及说明）  1.更新学生信息  （1）输入想要更新的学生学号，  （2）遍历整个链表，依次判断链表中的学号与目标学号是否匹配，若匹配，则转（3）；否则，提示“未找到该学号的学生”。  （3）选择需要更新的是德育分或体育分或智育分。  （4）键入更新后的分数，  （5）再次通过calculateComprehensiveScore(Student\* s)函数计算出新的综合分。  2.查询学生信息  （1）选择查询方式（按学号查询，按姓名查询，按专业查询）。  （2）遍历整个链表，依次判断链表中的（学号或姓名或专业）与目标（学号或姓名或专业）是否匹配，若匹配，则转（3）；否则，提示“未找到对应学生”。  （3）输出该学生的所有信息。  3.浏览所有学生信息  （1）提示“所有学生信息如下：”  （2）遍历整个链表，依次输出所有学生信息  4.统计不及格学生人数  （1）输入学院和专业  （2）遍历整个链表，依次找到符合该学院该专业且综合分小于60的学生，若找到，则转（3）后继续遍历  （3）计数+1  （4）输出所有不及格学生人数  **2、代码设计**  （对1、进行代码实现，在代码中关键操作都要有注释）  **1.更新学生信息**  // 更新学生信息  void updateStudentInfoByID() {  int studentID; // 学号  printf("请输入要更新的学生学号："); // 提示输入学号  scanf("%d", &studentID); // 输入学号  Student\* current = head; // 当前学生指针  while (current != NULL) { // 遍历链表  if (current->studentID == studentID) { // 如果找到学号匹配的学生  printf("请输入新的信息：\n"); // 提示输入新信息  printf("德育分: ");  scanf("%f", &current->moralEducationScore); // 输入新的德育分  printf("体育分: ");  scanf("%f", &current->physicalEducationScore); // 输入新的体育分  printf("智育分: ");  scanf("%f", &current->intellectualEducationScore); // 输入新的智育分  current->comprehensiveScore = calculateComprehensiveScore(current); // 计算新的综合分  printf("信息更新成功。\n"); // 提示更新成功  return;  }  current = current->next; // 更新当前学生指针  }  printf("未找到该学号的学生。\n"); // 提示未找到学生  }  2.查询学生信息  void queryStudentInfo() {  int choice; // 查询方式选择  printf("请选择查询方式：\n");  printf("1. 按学号查询\n");  printf("2. 按姓名查询\n");  printf("3. 按专业查询\n");  scanf("%d", &choice); // 输入选择  Student\* current = head; // 当前学生指针  switch (choice) {  case 1: {  int studentID; // 学号  printf("请输入学号："); // 提示输入学号  scanf("%d", &studentID); // 输入学号  while (current != NULL) { // 遍历链表  if (current->studentID == studentID) { // 如果找到学号匹配的学生  printf("查询结果：\n学号:%d 姓名:%s 性别:%s 出生日期:%s 专业:%s 所属学院:%s:%.2f:%.2f:%.2f:%.2f\n",current->studentID,current->name,current->gender, current->birthDate,current->major,current->college,current->moralEducationScore,current->physicalEducationScore,current->intellectualEducationScore, current->comprehensiveScore); // 输出学生信息  return;  }  current = current->next; // 更新当前学生指针  }  printf("未找到该学号的学生。\n"); // 提示未找到学生  break;  }  case 2: {  char queryStrname[50];  printf("请输入姓名：");  scanf("%s", queryStrname);  while (current != NULL) {  if (strcmp(current->name, queryStrname) == 0){  printf("查询结果：\n学号:%d 姓名:%s 性别:%s 出生日期:%s 专业:%s 所属学院:%s 德育分:%.2f 体育分:%.2f 智育分:%.2f 综合:%.2f\n",current->studentID,current->name,current->gender, current->birthDate,current->major,current->college,current->moralEducationScore, current->physicalEducationScore,current->intellectualEducationScore, current->comprehensiveScore);  current = current->next;  return;}  }  }  case 3: {  char queryStrmajor[50];  printf("请输入专业：");  scanf("%s", queryStrmajor);  while (current != NULL) {  if(strcmp(current->major, queryStrmajor) == 0) {  printf("查询结果：\n学号:%d 姓名:%s 性别:%s 出生日期:%s 专业:%s 所属学院:%s 德育分:%.2f 体育分:%.2f 智育分:%.2f 综合:%.2f\n",current->studentID,current->name,current->gender, current->birthDate,current->major,current->college,current->moralEducationScore, current->physicalEducationScore,current->intellectualEducationScore, current->comprehensiveScore);  }  current = current->next;  }  return;  }  default: printf("无效的选择。\n");  }  }  3.浏览学生信息  void browseAllStudents() {  printf("所有学生信息如下：\n"); // 提示浏览所有学生信息  Student\* current = head; // 当前学生指针  while (current != NULL) { // 遍历链表  printf("学号:%d 姓名:%s 性别:%s 出生日期:%s 专业:%s所属学院:%s 德育分:%.2f 体育分:%.2f 智育分:%.2f 综合分:%.2f\n",current->studentID, current->name,current->gender,current->birthDate,current->major,current->college,current->moralEducationScore,current->physicalEducationScore,current->intellectualEducationScore, current->comprehensiveScore); // 输出学生信息  current = current->next; // 更新当前学生指针  }  }  4.统计不及格学生人数  void countFailingStudentsInCollegeAndMajor() {  char college[50], Major[50]; // 学院和专业  printf("请输入学院和专业："); // 提示输入学院和专业  scanf("%s %s", college, Major); // 输入学院和专业  int count = 0; // 不及格学生人数  Student\* current = head; // 当前学生指针  while (current != NULL) { // 遍历链表  if (strcmp(current->college, college) == 0 && strcmp(current->MajorName, Major) == 0 && current->comprehensiveScore < 60){ // 如果找到匹配的学生且综合分小于60  count++; // 计数加1}  current = current->next; // 更新当前学生指针  }  printf("不及格学生人数：%d\n", count); // 输出不及格学生人数  } | | | | | | | | | |
| **实**  **践**  **总**  **结** | | （详述自己在本课题中所承担的全部工作；本课程设计完成了哪些功能还有哪些地方需要改进；调试程序中遇到的问题及采取的解决措施等；取得的收获、不足与建议等等，不少于500字，不接受发表感想之类的大话空话！）  我在本课题中主要负责：  更新学生信息，查询学生信息，浏览学生信息和统计不及格学生名单。本课程设计完成了添加、删除、更新、查询、浏览和统计学生信息，提供了清晰的菜单系统，让用户能够轻松地选择不同的操作，在每个步骤提供清晰的指示，  遇到的问题及措施：  在需要用户键入信息时，要确保占位符与要接收的数据格式正确（如int整形-%d，float浮点型-%f和string字符串类型-%s）  在查询switch（choice）的三种情况下，case1是按学号查询，case2是按姓名查询，case3是按专业查询，其中，case1与case2、case3情况不同，case1是通过输入正数，而case2，case3是通过输入字符串类型实现。所以最开始的想法是将case2与case3合并在一起实现，当姓名或专业其中一方满足要查找的条件时即可输出学生信息，但经过再三考虑，目的为了用户体验和实用性，使代码更加清晰和易于理解，还是决定将case2，case3拆分开，选择的case和要查找的条件分别满足后才输出学生信息。因为在题目要求中同时有专业和班级，考虑到其两者属于同一个范畴，可以合并，为了减少繁琐部分，并简化数据结构，故考虑将班级部分去除掉。用户输入时直接将专业班级同时输入即可。  仍有一些可改进的地方：  例如排序功能的运行，只显示“排序完成”，而不会显示出排序后详细的学生信息名单等；  取得的收获：  通过这次课题实践，经过许多次翻阅、研究课本及重温上学期的知识（例如switch选择菜单语句，自定义函数，链表等）感慨颇多。从制作到发现问题，经历无数次或大或小的修改解决问题，最后组成的一行行密密麻麻的代码都融入着我们滴滴心血于中，也让我看到自己的知识组成仍不够扎实与完备，最后以一句逍遥游的成语共勉——知不足而奋进，望远山而力行。真正付诸实践，才能不断获得成长。 | | | | | | | | | |
| **学号** | | | **2324105** | | | **姓名** | | | **张佳豪** | | |
| **详**  **细**  **设**  **计**  **及**  **实**  **施** | | **1、算法设计**  （对本人承担的各模块用传统流程图或自然语言描述及说明）  1.添加学生信息  （1）通过malloc分配内存来创建一个新的 Student 结构体  （2）然后提示用户输入学生的详细信息，包括学号、姓名、性别、出生日期、专业、班级、所属学院、德育分、体育分和智育分  （3）接着，它调用 calculateComprehensiveScore函数来计算综合分数，并将新节点添加到链表的头部  （4）最后，打印出一条消息表示学生信息已成功添加。  2删除学生信息   1. 首先，提示用户输入要删除的学生的学号 2. 然后遍历链表以查找具有该学号的学生节点 3. 如果找到了匹配的节点，将从链表中删除该节点并释放其内存 4. 如果没有找到匹配的节点，则打印一条消息表示未找到该学号的学生   3排序学生信息（运用了插入排序法）   1. 首先，检查链表是否为空或只有一个节点 2. 如果是，则不需要排序 3. 然后，遍历原始链表，并将每个节点插入到已排序链表的正确位置 4. 最后，将头指针更新为已排序链表的头部，并打印一条消息表示排序已完成   4.保存学生信息  （1）首先，尝试打开名为 "students.dat" 的文件  （2）如果文件无法打开，则打印一条错误消息并返回  （3）打开文件，然后，遍历链表并将每个学生节点的数据写入文件  （4）最后，关闭文件并打印一条消息表示数据已成功保存。  5.读取学生信息  （1）首先，尝试打开名为 "students.dat" 的文件  （2）如果文件无法打开，则打印一条错误消息并返回  （3）打开文件，然后，循环读取文件中的数据，每次读取一个 Student 结构体的大小，并将其分配到一个新的节点中  （4）如果成功读取了一个学生节点，将该节点添加到链表的头部；否则，释放已分配的内存  （5）当到达文件末尾时，关闭文件并打印一条消息表示数据已成功读取。  **2、代码设计**  （对1、进行代码实现，在代码中关键操作都要有注释）  **1.添加学生信息**  void addStudent() {  Student\* newStudent = (Student\*)malloc(sizeof(Student)); // 动态分配内存给新学生  printf("请输入学生信息：\n"); // 提示输入学生信息  printf("学号: ");  scanf("%d", &newStudent->studentID); // 输入学号  printf("姓名: ");  scanf("%s", newStudent->name); // 输入姓名  printf("性别: ");  scanf("%s", newStudent->gender); // 输入性别  printf("出生日期: ");  scanf("%s", newStudent->birthDate); // 输入出生日期  printf("专业: ");  scanf("%s", newStudent->major); // 输入专业  printf("班级: ");  scanf("%s", newStudent->className); // 输入班级  printf("所属学院: ");  scanf("%s", newStudent->college); // 输入所属学院  printf("德育分: ");  scanf("%f", &newStudent->moralEducationScore); // 输入德育分  printf("体育分: ");  scanf("%f", &newStudent->physicalEducationScore); // 输入体育分  printf("智育分: ");  scanf("%f", &newStudent->intellectualEducationScore); // 输入智育分  newStudent->comprehensiveScore = calculateComprehensiveScore(newStudent); // 计算综合分  newStudent->next = head; // 将新学生插入链表头  head = newStudent; // 更新链表头指针  printf("学生信息添加成功。\n"); // 提示添加成功  }  2. 删除学生  void deleteStudentByID() {  int studentID; // 学号  printf("请输入要删除的学生学号："); // 提示输入要删除的学号  scanf("%d", &studentID); // 输入学号  Student \*current = head, \*previous = NULL; // 当前学生和前一个学生指针  while (current != NULL) { // 遍历链表  if (current->studentID == studentID) { // 如果找到学号匹配的学生  if (previous == NULL) { // 如果是链表头  head = current->next; // 更新链表头指针  } else {  previous->next = current->next; // 前一个学生指向下一个学生  }  free(current); // 释放当前学生内存  printf("学生信息删除成功。\n"); // 提示删除成功  return;  }  previous = current; // 更新前一个学生指针  current = current->next; // 更新当前学生指针  }  printf("未找到该学号的学生。\n"); // 提示未找到学生  }  3.排序学生信息  void sortStudentsByID() {  if (head == NULL || head->next == NULL) return; // 如果链表为空或只有一个节点，直接返回  Student\* sorted = NULL; // 已排序的链表  Student\* current = head; // 当前学生指针  while (current != NULL) { // 遍历链表  Student\* next = current->next; // 下一个学生指针  if (sorted == NULL || sorted->studentID >= current->studentID) { // 如果已排序链表为空或当前学生学号小于或等于已排序链表头  current->next = sorted; // 将当前学生插入已排序链表头  sorted = current; // 更新已排序链z表头  } else {  Student\* temp = sorted; // 临时指针  while (temp->next != NULL && temp->next->studentID < current->studentID) { // 找到插入位置  temp = temp->next; // 更新临时指针  }  current->next = temp->next; // 将当前学生插入已排序链表  temp->next = current; // 更新已排序链表  }  current = next; // 更新当前学生指针  }  head = sorted; // 更新链表头指针  printf("排序完成。\n"); // 提示排序完成  }  4.保存数据到文件  void saveDataToFile() {  FILE \*file = fopen("students.dat", "wb"); // 打开文件  if (file == NULL) { // 如果文件打开失败  printf("无法打开文件。\n"); // 提示无法打开文件  return;  }  Student\* current = head; // 当前学生指针  while (current != NULL) { // 遍历链表  fwrite(current, sizeof(Student), 1, file); // 将学生数据写入文件  current = current->next; // 更新当前学生指针  }  fclose(file); // 关闭文件  printf("数据保存成功。\n"); // 提示数据保存成功  }  5.从文件读取数据  void loadDataFromFile() {  FILE \*file = fopen("students.dat", "rb"); // 打开文件  if (file == NULL) { // 如果文件打开失败  printf("无法打开文件。\n"); // 提示无法打开文件  return;  }  Student\* newStudent; // 新学生指针  while (!feof(file)) { // 循环读取文件  newStudent = (Student\*)malloc(sizeof(Student)); // 动态分配内存  if (fread(newStudent, sizeof(Student), 1, file) == 1) { // 如果读取成功  newStudent->next = head; // 将新学生插入链表头  head = newStudent; // 更新链表头指针  } else {  free(newStudent); // 释放内存  }  }  fclose(file); // 关闭文件  printf("数据读取成功。\n"); // 提示数据读取成功  } | | | | | | | | | |
| **实**  **践**  **总**  **结** | | （详述自己在本课题中所承担的全部工作；本课程设计完成了哪些功能还有哪些地方需要改进；调试程序中遇到的问题及采取的解决措施等；取得的收获、不足与建议等等，不少于500字，不接受发表感想之类的大话空话！）  **总结**  在本次学生考评信息管理系统的开发过程中，我主要负责了对学生信息进行添加、删除、保存和读取等关键功能的代码编写工作。  一、功能实现  1. 学生信息添加：通过设计友好的用户界面，获取用户输入的学生姓名、学号、成绩等信息，并将其准确无误地添加到系统数据库中。在实现过程中，我着重考虑了数据的合法性校验，确保输入的信息符合规定的格式和范围。  2. 学生信息删除：根据用户提供的学号或其他特定条件，能够准确地从数据库中删除相应的学生记录。为了防止误删除，还添加了确认提示功能。  3. 学生信息排序：实现了按照学生成绩、学号等字段进行升序排列的功能。这部分代码主要运用了插入排序法。  4. 学生信息保存：在用户进行信息添加、修改或删除操作后，能够及时将更新后的学生信息保存到数据库中，确保数据的完整性和一致性。  5. 学生信息读取：能够从数据库中准确读取学生信息，并将其展示在系统界面上，方便用户查看和操作。  二、已完成的功能  本课程设计基本完成了预期的主要功能，包括学生信息的增删改查以及排序操作。用户可以方便地添加新的学生记录，删除不再需要的记录，对学生信息进行排序以方便查看和分析，并且能够随时保存和读取数据，保证了系统的稳定性和数据的安全性。  三、需要改进的地方  在课程设计过程中，我也意识到系统存在一些不足之处。首先，界面设计还不够美观和人性化，操作流程可能对于一些用户来说不够简洁直观。其次，在数据处理的效率方面还有提升的空间，尤其是在处理大量数据时，系统的响应速度有待提高。  四、调试程序中遇到的问题及解决措施  在调试程序的过程中，我遇到了不少问题。例如，在进行数据保存时，出现了数据乱码的情况。经过仔细检查代码，发现是由于写入数据的模式不对导致的，二进制写入模式会导致有些数据读取不出来，就会导致乱码。  五、收获与不足  通过这次课程设计，我收获颇丰。首先，我对编程语言的理解和运用能力得到了很大的提升，尤其是在数据结构和算法的应用方面有了更深入的理解。其次，我学会了如何从需求分析到代码实现的整个软件开发流程，提高了自己的问题解决能力和逻辑思维能力。同时，我也深刻体会到了团队合作的重要性，在与同学的交流和协作中，学到了很多宝贵的经验和技巧。我也意识到自己还存在很多不足。在代码规范和注释方面做得不够好，导致代码的可读性较差。在面对复杂问题时，有时会感到无从下手，缺乏足够的耐心和细心。此外，对于一些新的技术和知识，还需要加强学习和探索。 | | | | | | | | | |
| **学号** | | | | |  | | | **姓名** | | |  |
| **详**  **细**  **设**  **计**  **及**  **实**  **施** | **1、算法设计**  （对本人承担的各模块用传统流程图或自然语言描述及说明）  **2、代码设计**  （对1、进行代码实现，在代码中关键操作都要有注释） | | | | | | | | | | |
| **实**  **践**  **总**  **结** | （详述自己在本课题中所承担的全部工作；本课程设计完成了哪些功能还有哪些地方需要改进；调试程序中遇到的问题及采取的解决措施等；取得的收获、不足与建议等等，不少于500字，不接受发表感想之类的大话空话！） | | | | | | | | | | |