

开发一个图书信息管理系统

河南工业大学

课程报告

课程名称： 程序设计实践

专业班级： 信息类 1107

学生姓名： 王春平

学号： 201116910730

任课教师： 阎娟

学 期： 2011-2012 学年第二学期

目 录	图书信息管理系统
-----	----------

主 要 内 容	<p>开发一个图书信息管理系统，图书信息包括：图书编号、书名、作者、出版社、类别、出版时间、价格等基本信息（也可以根据自己情况进行扩充，比如是否借出、库存量等）。使之能提供以下基本功能：</p> <p>(1) 图书信息录入功能(图书信息用文件保存)——输入？</p> <p>(2) 图书信息浏览功能——输出</p> <p>(3) 查询功能(至少一种查询方式)、排序功能(至少一种排序方式)：</p> <p>①按书名查询 ②按作者名查询 ③按照价钱排序 ④按出版时间排序等等</p> <p>(4) 图书信息的删除与修改</p> <p>扩展功能：可以按照自己的程度进行扩展。比如(1)简单的权限处理 (2)报表打印功能(3)甚至根据自己情况，可以加上学生信息，并扩充为图书借阅系统。(4)模糊查询 (5)综合查询 (6)统计功能 比如统计处某一类别的图书信息 或 筛选出小于指定数量库存的图书信息等等，总之，可以根据自己需求进行分析功能。</p>
------------------	---

任务要求	<p>一、提交材料应包括：(1) 系统源代码 (2) 课程报告</p> <p>二、整个设计过程具体要求</p> <p>(1) 需求分析 要求学生案例系统进行分析，设计出需要完成的功能，完善各个模块的调用关系；</p> <p>(2) 设计过程 要求学生进一步明确各模块调用关系，进一步完善模块函数细节（函数名、参数、返回值等）</p> <p>(3) 实现过程 要求学生养成良好的编码习惯、完成各个模块并进行测试，最终完成系统整体测试；</p> <p>(4) 总结阶段 按照要求完成系统设计和实现报告，并进行总结、答辩。</p>							
	报告撰写情况 (30 分)			系统完成情况 (30 分)		答辩情况 (40 分)		总分
	内容 20 分	规范程度 5 分	程序测试 5 分	基本功能 20 分	扩展功能 10 分	自述情况 10 分	答辩情况 30 分	

成绩评定教师：

课程报告任务书

1: 需求分析

1) 功能

创建一个图书管理系统；用文件保存和读出图书信息；

插入图书信息；

查询某本图书；

通过遍历输出所有图书信息；

删除与修改图书信息；

按照图书的某个信息对其进行排序；

统计图书系统中所含图书的数目；

设置进入图书管理系统的密码，使系统相对安全。

2) 需处理的数据

每本图书的信息：图书编号，书名，作者，出版社，类别，出版时间，价格，图书的剩余量。

图书信息用文本文件的形式保存；

2: 概要设计

(1) 数据结构

主要数据结构：链表

将每本图书的信息都保存在结构体中，并通过“结构体指针指向下一个结构体”的方式将图书信息连接成链表。每本图书信息保存的结构体如下：

```
struct list{
```

```

        int num;

        char name[20];

        char author[20];

        char publisher[20];

        char kind[20];

        char date;

    double price;

        int leaving;

        struct list * next;

};

```

(2) 模块划分

a. 创建链表 b. 插入 c. 删除——按图书编号删除 d 修改——按图书编号修改.e. 统计库存量 f. 遍历 g. 文件导入 h. 文件导出 i. 查询——按图书编号删除 j. 文件保存 k. 排序——按图书价格排序

(3) 程序总体框架

开始——>main 函数——>统计库存量 Length 函数, 创建 Creat 函数, 插入 Insert 函数, 删除 Delete 函数, 修改 Revise 函数, 遍历 Print 函数, 文件导入 fileread 函数, 文件导出 filewrite 函数, 查询 Search 函数, 文件保存 fprint 函数, 排序 Sorting 函数——>结束程序运行

(4) 函数简介

a: 创建链表 struct list * Create();

功能：创建图书信息函数来读取录入的图书信息，最后向 **main** 函数返回所建立链表的头指针。

参数：无

b: 插入图书信息 `struct list * Insert(struct list * head,struct list * book);`

功能：通过将创建函数和从 **main** 函数两种途径传过来的图书信息按编号连成一个链表，并将插入新元素的新链表的头指针返回给调用它的函数。最后 **main** 函数返回插入新元素后的链表的头指针。

参数:`struct list * head struct list * book`

c: 删除图书信息 `struct list * Delete(struct list * head,int num);`

功能：通过从 **main** 函数中得到图书信息链表的头指针，运用两个指针的依次移动分别找到要删除的图书信息元素的前一个元素和后一个元素，之后通过指针指向的变换达到删除元素的目的，最后返回 **main** 函数一个删除后图书信息链表的头指针。

参数： `struct list * head,int num`

d: 修改图书信息 `struct list * Revise(struct list * head,int num);`

功能：通过从 **main** 函数中得到图书信息链表的头指针，运用指针的移动将要修改的元素的编号依次与链表中元素的编号进行对比找到要修改信息的元素。最后返回 **main** 函数一个修改后的图书信息链表的头指针。

e: 图书信息遍历函数: `void Print(struct list * head);`

功能: 通过从 `main` 函数中得到图书信息链表的头指针, 通过设立的指针的依次向后移动输出每个图书的信息。

参数: `struct list * head`)

f: 图书信息统计函数 `int Length(struct list * head);`

功能: 通过对图书信息的遍历统计图书信息链表中共录入了多少图书信息, 最后返回 `main` 函数一个统计后的后图书信息链表的头指针。

参数: `struct list * head`

g: 图书信息输入函数 `struct list * fileread();`

功能: 从外部文件中读取保存在其中的图书信息, 并将这些信息建立成一个新的图书信息链表, 再把这个链表的头指针返回给 `main` 函数供其它函数使用

参数: 无

h: 图书信息输出函数 `void filewrite(struct list * head);`

功能: 将录入的图书信息按照链表中的信息顺序保存到文件中

参数: `struct list * head`

i: 对图书信息的排序 `struct list * Sorting(struct list * head);`

功能: 通过从 `main` 函数中得到图书信息链表的头指针, 再通过指针一次从链表中取出一个图书信息并把这些信息按选定的方式排序。最后返回 `main` 函数一个排序后图书信息链表的头指针。

参数: `struct list * head`

j: 保存图书信息 `void fprintf(struct list * head) ;`

功能：将图书信息永久的保存到文件中，并且在其它函数比如删除，修改中调用此函数，将处理后的图书信息保存到文件中

参数:struct list * head

3 详细设计

图书信息统计函数 int Length(struct list * head){

```
    int n=0;
```

```
    struct list * p;
```

```
    p=head;
```

```
    if(p==NULL)
```

```
        没有图书信息
```

```
    for(;p!=NULL;p=p->next)
```

```
        n++;
```

```
    return n;
```

```
}
```

图书信息创建函数

```
struct list * Create(){
```

```
    int num,leaving,q;
```

```
    char
```

```
name[20],author[20],publisher[20],kind[20],date[20];
```

```
    double price;
```

```
    int size=sizeof(struct list);
```

```
struct list * head,* p,*tail;
```

```
head=tail=NULL;
```

请输入图书编号

```
while(num!=0){
```

书名 作者 出版社 类别 出版时间 价格

剩余量

```
e,&price,&leaving);
```

```
p=(struct list *)malloc(size);
```

```
p->num=num;
```

```
strcpy(p->name,name);
```

```
strcpy(p->author,author);
```

```
strcpy(p->publisher,publisher);
```

```
strcpy(p->kind,kind);
```

```
p->price=price;
```

```
strcpy(p->date,date);
```

```
p->leaving=leaving;
```

```
p->next=NULL;
```

```
if(head==NULL)
```

```
head=p;
```

```
else
```

```
tail->next=p;
```

```
tail=p;
```

请输入图书编号

```
}
```

是否将图书信息保存至文件？（1-是 0-否）

```
if(q==1){
```

```
fprint(head);
```

图书信息已成功保存至 tushu_list.txt 中！

```
}
```

```
else
```

文件没有保存！

```
return head;
```

```
}
```

图书信息插入函数

```
struct list *Insert(struct list * head,struct list *  
book){
```

```
struct list *p,* p1,*p2;
```

```
int q;
```

```
p2=head;
```

```

p=book; /*p 指向待插入的结点*/

if(head==NULL)
{
    head=p; /* 新插入的结点成为头结点*/
    head->next=NULL;
}

else{
    while((p->num>p2->num)&&(p2->next!=NULL)){
        p1=p2; /*p1,p2 各后移一个结点*/
        p2=p2->next;
    }

    if(p->num<=p2->num) /* 在 p1,p2 之间插入新结点*/ {
        if(head==p2)
            head=p;
        else
            p1->next=p;
            p->next=p2;
    }

    else{
        p2->next=p;
        p->next=NULL;
    }
}

```

```
}
```

是否将插入新信息后的图书信息保存至文件？（1-

是 2-否）

```
if(q==1)
```

```
    fprintf(head);
```

添加图书信息成功

```
    return head;
```

```
}
```

图书信息删除函数

```
struct list * Delete(struct list * head,int num){
```

```
    struct list * p1,*p2;
```

```
    int z;
```

```
    while(head!=NULL&&head->num==num){
```

```
        p2=head;
```

```
        head=head->next;
```

```
        free(p2);
```

```
    }
```

```
    if(head==NULL)
```

```
        return NULL;
```

```
    p1=head;
```

```
    p2=head->next; /* 从表头的下一个结点搜索待删除的结点
```

*/

```
while(p2!=NULL){  
    if(p2->num==num)/*p2 所指结点符合删除要求*/{  
        p1->next=p2->next;  
        free(p2);  
    }  
    else  
        p1=p2; /*p1 后移一个结点*/  
        p2=p1->next; /*p2 指向 p1 的后一个结点*/  
}
```

是否将删除后的图书信息保存至文件？（1-是 2-否）

```
if(z==1)  
    fprintf(head);  
        删除图书信息成功  
return head;  
}
```

图书信息修改函数

```
struct list * Revise(struct list * head,int num){  
    struct list * p1,*p2;  
    int choice=0,d;
```

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/735210223330011304>