

南京航空航天大学

《面向对象程序设计语言》课程设计报告

网上书店管理系统

学号：161630216

姓名：李元成

日期：2017-4-22

目 录

一、基本信息：	错误!未定义书签。
二、试验形式与分工：	错误!未定义书签。
三、需求分析：	错误!未定义书签。
四、程序的主要功能：	错误!未定义书签。
五、系统总框架图	错误!未定义书签。
六、程序类的说明：	4
七、系统测试	错误!未定义书签。
八、比较有特色的函数	5
九、存在的不足与对策	5
十、使用说明	5
九、程序源代码	5

基本信息:

VC6.0 平台, c++程序

实验形式和分工

人员: 李元成

需求分析

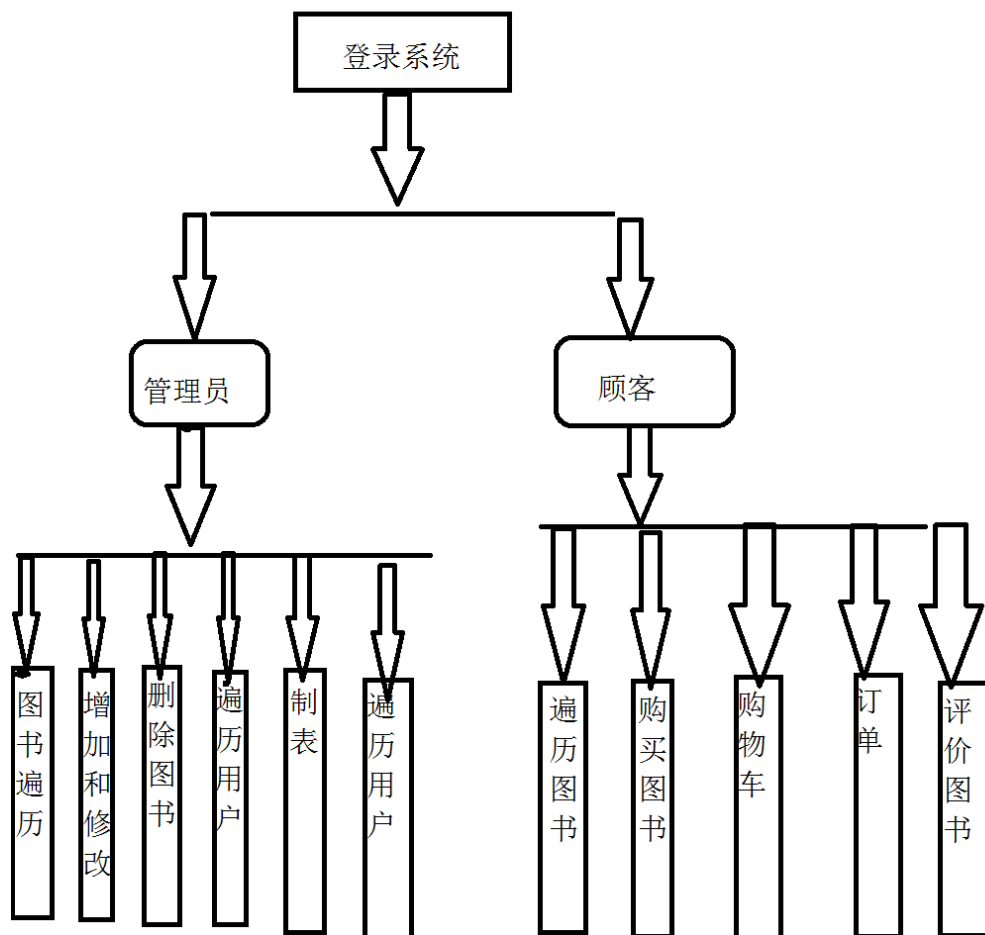
有两类人的需求, 一类是书店老板, 另一类是顾客。对于书店老板, 允许其遍历, 增加修改图书, 接受订单信息, 制表。对于顾客, 允许其遍历, 购买, 评价图书, 发送订单。

程序主要功能

登录。遍历, 修改, 购买, 评价图书。制表。

系统总框架图

在下一页



程序类的说明

```

class person
{protected:
    char ID[50];
    char key[50];
public:
    bool check(char a[50],char b[50])//登录;
    virtual void ls(){}//遍历图书

};
class adm :public person
{

```

```

public:
    adm();
    void ls();
    bool check(char a[50],char b[50]);
    void way();//菜单
};

class book
{private :
    int price;
    int sum;
    int ID;
    char name[50];
    char author[50];
    char publish[50];
    char brief[50];
};

void userls(); //顾客遍历
void del(); //删除图书
void add(); //添加
void change(); //修改
void check();
void load(); //加载图书
void bookwrite(); //往文件写书
void userload() ;
void useradd();
void userwrite();
void userlook(); //遍历用户
void comment(int); //评论
int search(char *a,int choice) //精准查找;
int search(int choice);
bool same(char ID[50]);
void seek(string a);//模糊查找
void tab();//制表
void vieworder();//查看订单

class order
{private:
    char ID[50];
    char name[50];
    int sum;
public:
    void set(char a[50],char b[50],int c);

```

系统测试

无

特色函数

Seek()模糊查找

利用 string 的 find()进行模糊查找

Tab()制表

图书编号和排序对象写入一个二维数组，利用冒泡法进行排序，
输出时候利用 search 获得图书信息

不足和对策

代码冗长，重复代码较多，可以写一个全局函数进行合并
继承的地方较少，应该再多发掘关系
加载重复，数据量大的时候效率不高，应该优化算法
防呆设计较少，错误的输入容易使程序崩溃

使用说明

进入按提示进入管理员和顾客系统
错误的输入容易是程序崩溃，请注意！
买书时候利用模糊查找获得精确的图书信息，然后利用图书信息进入精确查找，
就可以进入买书界面

程序源代码

```
adm
#include"StdAfx.h"
#include<iostream>
#include"person.h"
#include<string.h>
#include"globafunction.h"
#include"book.h"
using namespace std;
#include"adm.h"
extern book BOOK[1000];
extern int Booksum;
bool adm::check(char a[50],char b[50])
{{if(!strcmp(ID,a)&&!strcmp(key,b))
    return true;
    else
    return false;
```

```

    }
    }
    adm::adm()
    {strcpy(ID,"161630216");
     strcpy(key,"161630216");
    }

void adm::ls()
{
    load();
    int k=0;
    while(k!=Booksum)
    {cout<<"ID:"<<k<<endl;
     cout<<"书名:"<<BOOK[k].name<<endl<<"价格:"<<BOOK[k].price<<endl<<"总
数:"<<BOOK[k].sum<<endl;
     cout<<"作者:"<<BOOK[k].author<<endl<<"出版社:"<<BOOK[k].publish<<endl<<"简
介:"<<BOOK[k].brief<<endl;
     if(BOOK[k].sum==0)
     cout<<"该书库存为 0,请尽快补充库存!!!"<<endl;
     k++;
    }
    cout<<"共计:"<<Booksum<<endl;
}

void adm::way()
{start:
cout<<"1:遍历图书"<<endl;
cout<<"2:添加图书"<<endl;
cout<<"3:修改图书"<<endl;
cout<<"4:删除图书"<<endl;
cout<<"5:遍历用户"<<endl;
cout<<"6:制表"<<endl;
cout<<"7:订单"<<endl;
cout<<"8:退出"<<endl;
cout<<"choice:";
int choice;
cin>>choice;
if(choice==8)
exit(0);
if(choice==1)
ls();
else if(choice==2)
add();
else if(choice==3)
change();

```

```

else if(choice==4)
del();
else if(choice==5)
userlook();
else if(choice==6)
tab();
else if(choice==7)
vieworder();
goto start;
}

```

```

globalfunction.cpp
#include"StdAfx.h"
#include<iostream>
#include<fstream>
#include<string.h>
#include"adm.h"
#include"book.h"
#include"order.h"
#include"user.h"
#include"globalfunction.h"
using namespace std;
adm A;
book BOOK[1000];
user B[1000];
int Booksum=0;
int Usersum=0;
void load()
{fstream file;
Booksum=0;
file.open("book.txt",ios::in);
if(!file.is_open())
{
cout<<"打开失败!"<<endl;
A.way();
}
while(file.peek()!=EOF)
{file.read((char *)&BOOK[Booksum].price,sizeof(int));
file.read((char *)&BOOK[Booksum].sum,sizeof(int));
file.read(BOOK[Booksum].name,50*sizeof(char));
file.read(BOOK[Booksum].author,50*sizeof(char));
file.read(BOOK[Booksum].publish,50*sizeof(char));
file.read(BOOK[Booksum++].brief,50*sizeof(char));
}
}

```



```

file.close();
}

void usersl(void)
{
    load();
    fstream file;
    fstream com;
    com.open("comment.txt",ios::in);
    file.open("book.txt",ios::in);
    int k=0;
    while(k!=Booksum)
    {
        cout<<"ID:"<<k<<endl;
        cout<<"书名:"<<BOOK[k].name<<endl<<"价格:"<<BOOK[k].price<<endl<<"总
数:"<<BOOK[k].sum<<endl;
        cout<<"作者:"<<BOOK[k].author<<endl<<"出版社:"<<BOOK[k].publish<<endl<<"简
介:"<<BOOK[k].brief<<endl;
        k++;
    }
    cout<<"共计:"<<Booksum<<endl;
    file.close();
}

```

```

void add(void)
{
    fstream a;
    a.open("book.txt",ios::out|ios::app);
    int price,sum;
    char name[50],author[50],publish[50],biref[50];
    cout<<"单价:";
    cin>>price;
    cout<<"数量:";
    cin>>sum;
    cout<<"名字:";
    cin.get();
    cin.getline(name,50*sizeof(char));
    cout<<"作者:";
    cin.getline(author,50*sizeof(char));
    cout<<"出版社:";
    cin.getline(publish,50*sizeof(char));
    cout<<"简介:";
    cin.getline(biref,50*sizeof(char));
}

```

```

        a.write((char *)&price,sizeof(price));
        a.write((char *)&sum,sizeof(sum));
        a.write(name,50*sizeof(char));
        a.write(author,50*sizeof(char));
        a.write(publish,50*sizeof(char));
        a.write(biref,50*sizeof(char));
        a.close();
        cout<<"add success!\n";
        load();

    }

void change(void)
{ load();
  char name[50],author[50];
  int ID,k;
  char choice;
  cout<<"1)书名搜索\n2)作者搜索\n3)ID 搜索:";
  cin>>k;
  if(k==1)
  {cout<<"书名:";
   cin.get();
   cin.getline(name,50*sizeof(char));
   ID=search(name,k);
  }
  if(k==2)
  {cout<<"作者:";
   cin.get();
   cin.getline(author,50*sizeof(char));
   ID=search(author,k);
  }
  if(k==3)
  {cout<<"ID:";
   cin>>k;
   ID=search(k);
  }
  if(ID==-1)
  {cout<<"没有找到!"<<endl;
   A.way();
  }
  else
  {cout<<"名字:"<<BOOK[ID].name<<endl<<"价格:"<<BOOK[ID].price<<endl<<"总
数:"<<BOOK[ID].sum<<endl;
   cout<<"是否修改名字? (Y or N):";

```

```

        cin>>choice;
        if(choice=='y' || choice=='Y')
        {cout<<"修改后的名字:";
            cin.get();
            cin>>BOOK[ID].name;
        }
        cout<<"修改后的价格:";
        cin>>BOOK[ID].price;
        cout<<"修改后的总数:";
        cin>>BOOK[ID].sum;
    }

    fstream file;
    file.open("book.txt",ios::out);
    for(int j=0;j<Booksum;j++)
    {file.write((char *)&BOOK[j].price,sizeof(int));
        file.write((char *)&BOOK[j].sum,sizeof(int));
        file.write(BOOK[j].name,50*sizeof(char));
        file.write(BOOK[j].author,50*sizeof(char));
        file.write(BOOK[j].publish,50*sizeof(char));
        file.write(BOOK[j].brief,50*sizeof(char));
    }
    file.close();
    cout<<"change success!"<<endl;
    load();

}

void bookwrite(void)
{fstream file;
    file.open("book.txt",ios::out);
    for(int j=0;j<Booksum;j++)
    {file.write((char *)&BOOK[j].price,sizeof(int));
        file.write((char *)&BOOK[j].sum,sizeof(int));
        file.write(BOOK[j].name,50*sizeof(char));
        file.write(BOOK[j].author,50*sizeof(char));
        file.write(BOOK[j].publish,50*sizeof(char));
        file.write(BOOK[j].brief,50*sizeof(char));
    }
    file.close();
}

void del(void)
{load();
    int ID;
    fstream file;

```

```

int k;
char name[50],author[50];
cout<<"全部删除按 1,部分删除按 2:";
int choice;
cin>>choice;
if(choice==1)
{cout<<"输入 1 确认全部删除!!:";
  cin>>choice;
  if(choice==1)
  {
    file.open("book.txt",ios::out);
    file.close();
  }
}
if(choice==2)
{cout<<"1)书名搜索\n2)作者搜索\n3)ID 搜索:";
  cin>>k;
  if(k==1)
  {cout<<"书名:";
    cin.get();
    cin.getline(name,50*sizeof(char));
    ID=search(name,k);
  }
  if(k==2)
  {cout<<"作者:";
    cin.get();
    cin.getline(author,50*sizeof(char));
    ID=search(author,k);
  }
  if(k==3)
  {cout<<"ID:";
    cin>>k;
    ID=search(k);
  }
  if(ID==-1)
  cout<<"没有找到!"<<endl;

else
{cout<<BOOK[ID].name<<endl;
  BOOK[ID].sum=-1;
  file.open("book.txt",ios::out);
  k=0;
  while(k!=Booksum)
  {if(BOOK[k].sum!=-1)

```

```

        {file.write((char *)&BOOK[k].price,sizeof(int));
          file.write((char *)&BOOK[k].sum,sizeof(int));
          file.write(BOOK[k].name,50*sizeof(char));
          file.write(BOOK[k].author,50*sizeof(char));
          file.write(BOOK[k].publish,50*sizeof(char));
          file.write(BOOK[k].brief,50*sizeof(char));
        }
        k++;
      }
      cout<<"del success"<<endl;
    }
  } file.close();
  load();
}

int search(char *a,int choice)
{load();

```

```

  if(choice==1)
  {for(int j=0;j<Booksum;j++)
  {
    if(!strcmp(BOOK[j].name,a))
    return j;
  }
  return -1;
}

if(choice==2)
{for(int j=0;j<Booksum;j++)
{if(!strcmp(BOOK[j].author,a))
  return j;
}
return -1;
}

else
return -1;
}

int search(int choice)
{load();
  if(choice>=Booksum||choice<0)
  return -1;
  else
  return choice;
}

void userload()
{fstream file;

```

```

    Usersum=0;
file.open("user.txt",ios::in);
if(file.is_open())
{while(file.peek()!=EOF)
{file.read((char *)&B[Usersum].ID,50*sizeof(char));
file.read((char *)&B[Usersum].key,50*sizeof(char));
file.read((char *)&B[Usersum].history,2500*sizeof(char));
file.read((char *)&B[Usersum].buy,2500*sizeof(char));
file.read((char *)&B[Usersum].buyprice,50*sizeof(int));
file.read((char *)&B[Usersum].buysum,50*sizeof(int));
file.read((char *)&B[Usersum].number,sizeof(int));
    Usersum++;
}
}
file.close();
}
void useradd()
{ fstream file;
file.open("user.txt",ios::out|ios::app);
file.write((char *)&B[Usersum].ID,50*sizeof(char));
file.write((char *)&B[Usersum].key,50*sizeof(char));
file.write((char *)&B[Usersum].history,2500*sizeof(char));
file.write((char *)&B[Usersum].buy,2500*sizeof(char));
file.write((char *)&B[Usersum].buyprice,50*sizeof(int));
file.write((char *)&B[Usersum].buysum,50*sizeof(int));
file.write((char *)&B[Usersum].number,sizeof(int));
    Usersum++;
file.close();
}
void userwrite()
{
fstream file;
file.open("user.txt",ios::out);
for(int k=0;k<Usersum;k++)
{ file.write((char *)&B[k].ID,50*sizeof(char));
file.write((char *)&B[k].key,50*sizeof(char));
file.write((char *)&B[k].history,2500*sizeof(char));
file.write((char *)&B[k].buy,2500*sizeof(char));
file.write((char *)&B[k].buyprice,50*sizeof(int));
file.write((char *)&B[k].buysum,50*sizeof(int));
file.write((char *)&B[k].number,sizeof(int));
}
file.close();
userload();

```

```

}
void userlook()
{userload();
  for(int j=0;j<Usersum;j++)
    cout<<"ID:"<<B[j].ID<<endl;
  cout<<"共计"<<Usersum<<"个用户"<<endl;
}
bool same(char ID[50])
{userload();
  for(int j=0;j<Usersum;j++)
    if(!strcmp(ID,B[j].ID))
      return true;
  return false;
}
void comment(int ID)
{fstream file;
  file.open("comment.txt",ios::in);
  char name[50];
  char comment[1000];
  cout<<"评论区:"<<endl;
  while(file.peek()!=EOF)
  {file.read(name,50*sizeof(char));
   file.read(comment,1000*sizeof(char));
   if(!strcmp(name,BOOK[ID].name))
     cout<<comment<<endl;
  }
}
void seek(string a)
{load();
  string b;
  cout<<"在书名中查找中..."<<endl;
  for(int k=0;k<Booksum;k++)
  {
    b=BOOK[k].name;
    if(b.find(a)!=string::npos)
      cout<<"ID:"<<k<<endl<<"书名:"<<BOOK[k].name<<endl<<"作者:"<<BOOK[k].author<<endl<<endl;
  }
  cout<<"在作者中查找..."<<endl;
  for(int l=0;l<Booksum;l++)
  {
    b=BOOK[l].author;
    if(b.find(a)!=string::npos)

```

```

        cout<<"ID:"<<l<<endl<<"书名:"<<BOOK[l].name<<endl<<"作者:"<<BOOK[l].author<<endl<<endl;
    }
    cout<<"在出版社中查找..."<<endl;
    for(int m=0;m<Booksum;m++)
    {
        b=BOOK[m].publish;
        if(b.find(a)!=string::npos)
            cout<<"ID:"<<m<<endl<<"书名:"<<BOOK[m].name<<endl<<"作者:"<<BOOK[m].author<<endl<<endl;
    }
}

void tab()
{load();
    cout<<"1)售价制表 2)数量制表 3)销量制表"<<endl;
    int choice;
    cin>>choice;
    if(choice==1)
    {int temp[1000][2];
        int t;
        for(int k=0;k<Booksum;k++)
        {temp[k][0]=k;
            temp[k][1]=BOOK[k].price;
        }
        for(int i=0;i<Booksum;i++)
        for(int j=0;j<Booksum-k-1;j++)
        {if(temp[j][1]<temp[j+1][1])
            {t=temp[j][1];
                temp[j][1]=temp[j+1][1];
                temp[j+1][1]=t;

                t=temp[j][0];
                temp[j][0]=temp[j+1][0];
                temp[j+1][0]=t;
            }
        }
        for(int l=0;l<Booksum;l++)
        for(int j=0;j<Booksum;j++)
        {if(j==temp[l][0])
            {cout<<l+1<<":"<<BOOK[j].name<<"      价格:"<<BOOK[j].price<<endl;
                break;
            }
        }
    }
}

```



```

else if(choice==2)
{int temp[1000][2];
  int t;
  for(int k=0;k<Booksum;k++)
  {temp[k][0]=k;
    temp[k][1]=BOOK[k].sum;
  }
  for(int m=0;m<Booksum;m++)
  for(int j=0;j<Booksum-m-1;j++)
  {if(temp[j][1]<temp[j+1][1])
    {t=temp[j][1];
      temp[j][1]=temp[j+1][1];
      temp[j+1][1]=t;

      t=temp[j][0];
      temp[j][0]=temp[j+1][0];
      temp[j+1][0]=t;
    }
  }
  for(int n=0;n<Booksum;n++)
  for(int j=0;j<Booksum;j++)
  {if(j==temp[n][0])
    cout<<n+1<<":"<<BOOK[j].name<<"      数量:"<<BOOK[j].sum<<endl;
  }
}

else if(choice==3)
{fstream file;
  file.open("order.txt",ios::in);
  int sum[1000][2];
  int ID,temp,num=0;
  int k;
  for(int l=0;l<10;l++)
  {sum[l][0]=-1;
    sum[l][1]=0;
  }
  bool flag=false;
  char b[50],a[50];
  while(file.peek()!=-1)
  {file.read(b,50*sizeof(char));
    file.read(a,50*sizeof(char));
    for(k=0;k<10;k++)
    {if(sum[k][0]==search(a,1))
      {
        flag=true;

```

```

        break;
    }
}
file.read((char *)&temp,sizeof(int));
if(flag)
    sum[k][1]+=temp;
else
{
    sum[num][0]=search(a,1);
    sum[num][1]+=temp;
    num++;
}
flag=false;
}
for( k=0;k<num;k++)
for(int j=0;j<num-k-1;j++)
{
    if(sum[j][1]<sum[j+1][1])
    {
        temp=sum[j][0];
        sum[j][0]=sum[j+1][0];
        sum[j+1][0]=temp;

        temp=sum[j][1];
        sum[j][1]=sum[j+1][1];
        sum[j+1][1]=temp;
    }
}
for(int n=0;n<num;n++)
{
    ID=search(sum[n][0]);
    strcpy(a,BOOK[ID].name);
    cout<<a<<"    "<<sum[n][1]<<"本"<<endl;
}
file.close();
}

}

```

```

void vieworder()
{
    fstream file;
    file.open("order.txt",ios::in);
    char ID[50],name[50];
    int sum;
    while(file.peek()!=-1)
    {
        file.read(ID,50*sizeof(char));
        file.read(name,50*sizeof(char));
    }
}

```

```

        file.read((char *)&sum,sizeof(int));
        cout<<"ID:"<<ID<<"    书名:"<<name<<"    数量"<<sum<<endl;
    }
    file.close();
}

```

Main.cpp

```

#include"StdAfx.h"
#include <iostream>
#include<fstream>
#include<string.h>
#include"globalfunction.h"
#include"adm.h"
#include"book.h"
#include"order.h"
#include"user.h"
extern user B[1000];
extern int Usersum;
extern adm A;
int main(int argc, char** argv) {
    int choice,k;
    char ID[50];
    char key[50];
    char userID[50];
    char userkey[50];
    cout<<"1)顾客\n"<<"2)管理员"<<endl<<"登录为:";
    cin>>choice;
    cin.get();
    if(choice==1)
    {userload();
        cout<<"1)登陆 2)注册:";
        cin>>choice;
        cin.get();
        if(choice==1)
        {step1:
            cout<<"账号:";
            cin>>userID;
            cout<<"密码:";
            cin>>userkey;
            for(k=0;k<Usersum;k++)
            {if(B[k].check(userID,userkey))
                {cout<<"登陆成功!"<<endl;
                    break;
                }
            }
        }
    }
}

```

```

    }
    if(k==Usersum)
    {
        cout<<"账号与密码不匹配!"<<endl;
        goto step1;
    }
    B[k].way();
}
else if(choice==2)
{ step2:
    cout<<"账号:";
    cin>>userID;
    if(userID[0]<='9'&&userID[0]>='0')
    {cout<<"首位不能为数字!"<<endl;
        goto step2;
    }
    if(same(userID))
    {cout<<"该账号已经注册!"<<endl;
        goto step2;
    }
    cout<<"密码:";
    cin>>userkey;
    B[Usersum].create(userID,userkey);
    useradd();
    cout<<"注册成功!"<<endl;
    B[Usersum-1].way();
}
}
else if(choice==2)
{step3:
    cout<<"账号:";
    cin>>ID;
    cout<<"密码:";
    cin>>key;
    if(A.check(ID,key))
    A.way();
    else
    {
        cout<<"账号与密码不匹配!";
        goto step3;
    }
}
return 0;
}

```

Order.cpp

```
#include"StdAfx.h"
#include"order.h"
#include<string.h>
void order::set(char a[50],char b[50],int c)
{strcpy(ID,a);
  strcpy(name,b);
  sum=c;
}
```

Person.cpp

```
#include"StdAfx.h"
#include<iostream>
#include<string.h>
using namespace std;
#include"person.h"
bool person::check(char a[50],char b[50])
{{if(!strcmp(ID,a)&&!strcmp(key,b))
    return true;
    else
    return false;
}
}
```

User.cpp

```
#include"StdAfx.h"
#include"globalfunction.h"
#include<string>
#include<string.h>
#include"book.h"
#include<fstream>
#include"adm.h"
#include"stdlib.h"
extern adm A;
extern book BOOK[1000];
extern user B[1000];
extern int Booksum;
extern int Usersum;
bool user::check(char a[50],char b[50])
{if(!strcmp(ID,a)&&!strcmp(key,b))
    return true;
    else
    return false;
}
```

```

}
void user::create(char a[50],char b[50])
{strcpy(ID,a);
    strcpy(key,b);
    memset(history,0,2500*sizeof(char));
    memset(buy,0,2500*sizeof(char));
    for(int k=0;k<50;k++)
    {buyprice[k]=0;
        buysum[k]=0;
    }
    number=0;
}
void user::way()
{
    int choice;
    int k;
    start:
    cout<<"1)遍历图书 2)购买图书 3)我的购物车 4)我的订单 5)评价图书 6)退
出:";

    cin>>choice;
    cin.get();
    if(choice!=0)
    {
        if(choice==1)
        {userls();
        }
        else if(choice==2)
        {char name[50],author[50];
            int ID;
            step:
            cout<<"1)精准查找 2)模糊查找:";
            cin>>k;
            if(k==1)
            {
                cout<<"1)书名搜索\n2)作者搜索\n3)ID 搜索:";
                cin>>k;
                if(k==1)
                {cout<<"书名:";
                    cin.get();
                    cin.getline(name,50*sizeof(char));
                    ID=search(name,k);
                }
                if(k==2)
                {cout<<"作者:";

```

```

cin.get();
cin.getline(author,50*sizeof(char));
ID=search(author,k);
}
if(k==3)
{cout<<"ID:";
  cin>>k;
  ID=search(k);
}
if(ID==-1)
{cout<<"没有找到!"<<endl;
  goto step;
}
else
{
  cout<<"书名:"<<BOOK[ID].name<<endl<<"价格:"<<BOOK[ID].price<<endl<<"总
数:"<<BOOK[ID].sum<<endl;
  cout<<"作者:"<<BOOK[ID].author<<endl<<"出版
社:"<<BOOK[ID].publish<<endl<<"简介:"<<BOOK[ID].brief<<endl;
  comment(ID);
  cout<<"您是否想购买此书\n1)把此书添加到购物车 2)离开:";
  cin>>choice;
  if(choice==1)
  {int a;
    cout<<"购买数量:";
    cin>>a;
    if(a>BOOK[ID].sum)
    {cout<<"您购买的数量大于库存,无法购买!"<<endl;
      goto start;
    }
    if(a<0)
    {cout<<"错误的数量!"<<endl;
      goto start;
    }

    strcpy(buy[number],BOOK[ID].name);
    buyprice[number]=BOOK[ID].price;
    buysum[number]=a;
    number++;
    cout<<"添加成功!"<<endl;

  }
  userwrite();
  goto start;
}

```

```

    }
}
else if(k==2)
{cout<<"输入查找内容:";
  string enter;
  cin>>enter;
  seek(enter);
  goto start;
}
}
else if(choice==3)
{int b=0;
  for(int k=0;k<number;k++)
  {cout<<buy[k]<<"    单价:"<<buyprice[k]<<"    数量:"<<buysum[k]<<endl;
    b+=buyprice[k]*buysum[k];
  }
  cout<<"小计:"<<b<<"元"<<endl;
  if(b!=0)
  {
    cout<<"购买?(Y/N):";
    char choice;
    cin>>choice;
    if(choice=='y' || choice=='Y')
    {fstream file;
      file.open("order.txt",ios::out | ios::app);
      int historynum=0;
      for(historynum;history[historynum][0]!=0;historynum++)
      ;

      for(int k=historynum;k<historynum+number;k++)
      strcpy(history[k],buy[k-historynum]);

      for(int l=0;l<number;l++)
      {file.write(ID,50*sizeof(char));
        file.write(buy[l],50*sizeof(char));
        file.write((char *)&buysum[l],sizeof(int));
      }

      file.close();
      int ID;
      for(int m=0;m<number;m++)
      {ID=search(buy[m],1);

```



```

        BOOK[ID].sum-=buysum[m];
        buyprice[m]=0;
        buysum[m]=0;
    }
    memset(buy,0,2500*sizeof(int));
    number=0;
    cout<<"感谢您的购买!"<<endl;
    bookwrite();
}
userwrite();
}

    goto start;
}
else if(choice==4)
{for(int k=0;history[k][0]!=0;k++)
    cout<<k+1<<": "<<history[k]<<endl;
    if(history[0][0]==0)
        cout<<"您的购买历史为空!"<<endl;
    goto start;
}
else if(choice==5)
{char name[50],author[50];
    int ID;

    cout<<"1)书名搜索\n2)作者搜索\n3)ID 搜索:";
    cin>>k;
    if(k==1)
    {cout<<"书名:";
        cin.get();
        cin.getline(name,50*sizeof(char));
        ID=search(name,k);
    }
    if(k==2)
    {cout<<"作者:";
        cin.get();
        cin.getline(author,50*sizeof(char));
        ID=search(author,k);
    }
    if(k==3)
    {cout<<"ID:";
        cin>>k;
        ID=search(k);
    }
    if(ID==-1)

```

```

        {cout<<"没有找到!"<<endl;
          way();
        }
    else
    {cout<<"您要评价的书是:"<<BOOK[ID].name<<endl;
      int power=0;
      for(int k=0;history[k][0]!=0;k++)
      if(!strcmp(history[k],BOOK[ID].name))
      {power=1;
        break;
      }
      if(power==0)
      {cout<<"您没有购买过此书,无权评论!"<<endl;
        goto start;
      }
      else
      {cout<<"请输入您的评论:";
        char comment[1000];
        cin>>comment;
        fstream file;
        file.open("comment.txt",ios::out|ios::app);
        file.write(BOOK[ID].name,50*sizeof(char));
        file.write(comment,1000*sizeof(char));
        file.close();
        goto start;
      }
    }
    }
    else if(choice==6)
    exit(0);

    goto start;
}
}

```