图书管理系统 ：

代码：

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#define N 30

struct member /\*定义会员的结构体\*/

{

int num;

char name[N];

int points;

struct member \*next;

};

struct book /\*定义书籍结构体\*/

{

int num;

char name[N];

int price;

struct book \*next;

};

void DenLu(char \*user,char \*password); /\*登录页面\*/

int CompareUP(char \*user,char \*password); /\*验证用户输入的用户名及密码是否正确\*/

void PrintSM(void); /\*打印主菜单\*/

void MemberManagement (struct member \*head,int \*sum); /\*会员管理\*/

void PrintVM(void); /\*打印会员管理子菜单\*/

struct member\* ExistingMember(struct member \*head,int \*sum); /\*将已有会员信息写入 \*/

void PrintVMember (struct member \*head); /\*打印会员信息\*/

struct member\* NewMember(struct member \*head,int \*sum); /\*新建会员信息\*/

struct member\* ChangeMember(struct member \*head); /\*修改会员信息\*/

struct book\* ExistingBook(struct book \*head,int \*sum); /\*录入已有书籍信息\*/

struct book\* BookManagement (struct book \*head2,int \*sum); /\*书籍管理\*/

void PrintBM(void); /\*打印书籍管理子菜单\*/

void PrintBook (struct book \*head); /\*打印书籍信息\*/

struct book\* NewBook(struct book \*head,int \*sum); /\*新建书籍信息\*/

struct book\* DelBook(struct book \*head); /\*删除书籍\*/

void CheckOut(struct member \*head1,struct book \*head2); /\*购物结账\*/

void FereMemory(struct member \*head1,struct book \*head2);

int main()

{

int a;

char user[N],password[N]; /\*用户输入的用户名和密码\*/

int choice; /\*菜单跳转的选项\*/

int sum1=0; /\*会员总数\*/

int sum2=0; /\*书籍总数\*/

struct member \*head1=NULL;

struct book \*head2=NULL;

int i;

for (i=0; i<9; i++)

{

head1=ExistingMember(head1,&sum1); /\*录入已有会员信息\*/

}

for (i=0; i<9; i++)

{

head2=ExistingBook(head2,&sum2); /\*录入已有书籍信息\*/

}

system("cls");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\* 图书销售管理系统 \*\n");

printf("\* 1.登录系统 \*\n");

printf("\* 2.退出 \*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("请选择,输入数字(1-2):");

scanf("%d",&a);

if(a==1)

{

DenLu(user,password);

for(;;)

{

PrintSM();

scanf("%d",&choice);

switch (choice)

{

case 1:

MemberManagement(head1,&sum1);

break;

case 2:

head2=BookManagement(head2,&sum2);

break;

case 3:

CheckOut(head1,head2);

break;

case 4:

printf("注销成功！\n");

break;

}

if (choice==4) break;

}

FereMemory(head1,head2);

exit(0);

}

if(a!=1)

{ printf("欢迎再次使用本系统 !");

exit(0); /\*正常退出程序\*/

}

}

void DenLu (char user[N],char password[N])

{

int i=0,flag=0;

do

{

printf("请输入账号:");

scanf("%s",user);

printf("请输入密码:");

scanf("%s",password);

flag=CompareUP(user,password);

if (flag==1)

{

break;

}

else

{

i++;

printf("您输入的用户名或密码错误!您还有%d次尝试机会.\n",3-i);

}

}while(i<3);

if (flag==0)

{

system("cls");

printf("您已经输入了三次错误密码。\n");

printf("您已经退出系统.");

exit(0);

}

}

int CompareUP(char \*user,char \*password)/\*验证用户输入的用户名及密码是否正确的函数\*/

{

char User[N]="admin";

char Password[N]="123456"; /\*正确的用户名和密码\*/

if (strcmp(user,User)==0&&strcmp(password,Password)==0)

return 1;

else

return 0;

}

/\*打印菜单\*/

void PrintSM(void)

{

system("cls");

printf("登录成功！\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\* 图书销售管理系统 \*\n");

printf("\* 1.会员管理 \*\n");

printf("\* 2.书籍管理 \*\n");

printf("\* 3.购物结算 \*\n");

printf("\* 4.注销 \*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("请选择，输入数字（1-4）:");

}

/\*会员管理相关函数\*/

/\*录入已有会员信息\*/

struct member\* ExistingMember(struct member \*head,int \*sum)

{

struct member \*p=NULL,\*pr=head;

p=(struct member\*)malloc(sizeof(struct member));

if (p==NULL)

{

printf("内存不足。\n");

exit(0);

}

if (head==NULL)

{

head=p;

}

else

{

while (pr->next!=NULL)

{

pr=pr->next;

}

pr->next=p;

}

(\*sum)++;

p->num=\*sum;

switch (\*sum)

{

case 1:

strcpy(p->name,"张三");

p->points=100;

break;

case 2:

strcpy(p->name,"李四");

p->points=600;

break;

case 3:

strcpy(p->name,"王五");

p->points=500;

break;

case 4:

strcpy(p->name,"陆六");

p->points=400;

break;

case 5:

strcpy(p->name,"钱七");

p->points=900;

break;

case 6:

strcpy(p->name,"老八");

p->points=100;

break;

case 7:

strcpy(p->name,"阿伟");

p->points=100;

break;

case 8:

strcpy(p->name,"健康哥");

p->points=600;

break;

case 9:

strcpy(p->name,"美丽哥");

p->points=500;

break;

default:

printf("录入会员信息错误。\n");

}

p->next=NULL;

return head;

}

/\*会员管理\*/

void MemberManagement (struct member \*head1,int \*sum)

{

int choice;

system("cls");

while(1)

{

PrintVM();

scanf("%d",&choice);

switch (choice)

{

case 1:

PrintVMember(head1);

break;

case 2:

head1=NewMember(head1,sum);

break;

case 3:

head1=ChangeMember(head1);

break;

default:

break;

}

if (choice==4) break;

}

}

/\*打印会员管理子菜单\*/

void PrintVM(void)

{

system("cls");

printf("图书销售管理系统>会员管理\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\* 1.显示所有会员 \*\n");

printf("\* 2.添加会员信息 \*\n");

printf("\* 3.修改会员信息 \*\n");

printf("\* 4.返回主菜单 \*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("请选择，输入数字（1-4）:");

}

/\*新建会员信息\*/

struct member\* NewMember(struct member \*head,int \*sum)

{

struct member \*p=NULL,\*pr=head;

char name[N];

int points;

system("cls");

printf("图书销售管理系统 > 会员信息管理 > 添加会员信息\n");

p=(struct member\*)malloc(sizeof(struct member));

if (p==NULL)

{

printf("内存不足。\n");

exit(0);

}

if (head==NULL)

{

head=p;

}

else

{

while (pr->next!=NULL)

{

pr=pr->next;

}

pr->next=p;

}

(\*sum)++;

p->num=\*sum;

printf("请输入新会员姓名:\n");

scanf("%s",name);

printf("请输入新会员积分:\n");

scanf("%d",&points);

strcpy(p->name,name);

p->points=points;

p->next=NULL;

printf("添加成功!\n");

printf("按任意键返回.");

getch();

return head;

}

/\*打印现有会员信息\*/

void PrintVMember (struct member \*head)

{

struct member \*p=head;

system("cls");

printf("图书销售管理系统 > 会员信息管理 > 显示会员信息\n");

printf(" 会员号 \t姓名\t\t积分\n");

do

{

printf("\t%d\t%s\t\t%d\n",p->num,p->name,p->points);

p=p->next;

}

while(p!=NULL);

printf("按任意键返回.");

getch();

}

/\*修改会员信息\*/

struct member\* ChangeMember(struct member \*head)

{

system("cls");

printf("图书销售管理系统 > 会员信息管理 >修改会员信息\n");

int number,points; /\*用户输入需要修改的会员信息\*/

char name[N];

struct member \*p=head;

int flag=0;

do

{

printf("请输入要修改的会员号:");

scanf("%d",&number);

p=head;

while(p!=NULL)

{

if (p->num==number)

{

flag=1;

break;

}

p=p->next;

}

if (flag==0)

{

printf("没找到该会员，请重新输入会员号。\n");

}

else if (flag==1)

{

printf("请输入要修改的会员姓名:");

scanf("%s",name);

printf("请输入要修改的会员积分:");

scanf("%d",&points);

}

}while(flag!=1);

strcpy(p->name,name);

p->points=points;

printf("修改成功！\n");

printf("按任意键返回.");

getch();

return head;

}

/\*书籍管理相关函数\*/

/\*录入已有书籍信息\*/

struct book\* ExistingBook(struct book \*head,int \*sum)

{

struct book \*p=NULL,\*pr=head;

p=(struct book\*)malloc(sizeof(struct book));

if (p==NULL)

{

printf("内存不足。\n");

exit(0);

}

if (head==NULL)

{

head=p;

}

else

{

while (pr->next!=NULL)

{

pr=pr->next;

}

pr->next=p;

}

(\*sum)++;

p->num=\*sum;

switch ((\*sum))

{

case 1:

strcpy(p->name,"C#2.0宝典");

p->price=88;

break;

case 2:

strcpy(p->name,"Java编程基础");

p->price=55;

break;

case 3:

strcpy(p->name,"J2SE桌面应用程序开发");

p->price=60;

break;

case 4:

strcpy(p->name,"龙族");

p->price=45;

break;

case 5:

strcpy(p->name,"历史的温度");

p->price=55;

break;

case 6:

strcpy(p->name,"高等数学");

p->price=68;

break;

case 7:

strcpy(p->name,"活着");

p->price=78;

break;

case 8:

strcpy(p->name,"自控力");

p->price=29;

break;

case 9:

strcpy(p->name,"人间失格");

p->price=38;

break;

default:

printf("录入书籍信息错误。\n");

}

p->next=NULL;

return head;

}

/\*书籍管理\*/

struct book\* BookManagement (struct book \*head2,int \*sum)

{

int choice;

system("cls");

while(1)

{

PrintBM();

scanf("%d",&choice);

switch (choice)

{

case 1:

system("cls");

PrintBook(head2);

printf("按任意键返回.");

getch();

break;

case 2:

head2=NewBook(head2,sum);

break;

case 3:

head2=DelBook(head2);

break;

default:

break;

}

if (choice==4) break;

}

return head2;

}

/\*打印书籍管理子菜单\*/

void PrintBM(void)

{

system("cls");

printf("图书销售管理系统 >书籍管理\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\* 1.查看书目 \*\n");

printf("\* 2.添加书籍 \*\n");

printf("\* 3.删除书籍 \*\n");

printf("\* 4.返回主菜单 \*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("请选择，输入数字（1-4）:");

}

/\*新建书籍信息\*/

struct book\* NewBook(struct book \*head,int \*sum)

{

struct book \*p=NULL,\*pr=head;

char name[N];

int price;

system("cls");

printf("图书销售管理系统 > 会员信息管理 > 新增会员信息\n");

p=(struct book\*)malloc(sizeof(struct book));

if (p==NULL)

{

printf("内存不足。\n");

exit(0);

}

if (head==NULL)

{

head=p;

}

else

{

while (pr->next!=NULL)

{

pr=pr->next;

}

pr->next=p;

}

(\*sum)++;

p->num=\*sum;

printf("请输入图书名：");

scanf("%s",name);

printf("请输入图书价钱：");

scanf("%d",&price);

strcpy(p->name,name);

p->price=price;

p->next=NULL;

printf("添加成功!\n");

printf("按任意键返回.");

getch();

return head;

}

/\*打印现有书籍信息\*/

void PrintBook (struct book \*head)

{

struct book \*p=head;

printf("图书销售管理系统>书籍管理>书籍列表\n");

printf("图书号 图书名 图书价钱\n");

do

{

printf(" %d\t %-25s%5d\n",p->num,p->name,p->price);

p=p->next;

}

while(p!=NULL);

}

/\*删除书籍\*/

struct book\* DelBook(struct book \*head)

{

system("cls");

printf("图书销售管理系统 >书籍管理>书籍下架\n");

struct book \*p=head,\*pr=head;

char bookname[N]; /\*用户输入的图书名\*/

int flag=0;

do{

printf("请输入要删除的图书书名：");

scanf("%s",bookname);

p=head;

while (p!=NULL)

{

if (strcmp(p->name,bookname)==0)

{

flag=1;

printf("找到该图书，位置为：%d\n",p->num);

break;

}

p=p->next;

}

if (flag==0)

{

printf("没找到该图书，请重新输入图书名。\n");

}

}while(flag!=1);

if (p==head)

{

head=p->next;

free(p);

}

else

{

while (pr->next!=p)

{

pr=pr->next;

}

pr->next=p->next;

free(p);

}

printf("删除成功！\n");

printf("按任意键返回.");

getch();

return head;

}

/\*购物结账相关的函数\*/

/\*购物结账\*/

void CheckOut(struct member \*head1,struct book \*head2)

{

system("cls");

printf("图书销售管理系统 >购物结算\n");

PrintBook(head2);

int MemberNum; /\*用户会员号\*/

int BookNum; /\*用户购买书本的信息\*/

int amount;

int total;

struct member \*pm=head1;

struct book \*pb=head2;

while(1)

{

int flag1=0,flag2=0;

do

{

pm=head1;

printf("请输入会员号：");

scanf("%d",&MemberNum);

while(pm!=NULL)

{

if (pm->num==MemberNum)

{

flag1=1;

break;

}

pm=pm->next;

}

if (flag1==0)

{

printf("没有找到该会员，请重新输入会员号。\n");

}

}

while(flag1!=1);

do

{

pb=head2;

printf("请输入要购买的书籍编号：");

scanf("%d",&BookNum);

while(pb!=NULL)

{

if (pb->num==BookNum)

{

flag2=1;

break;

}

pb=pb->next;

}

if (flag2==0)

{

printf("没有找到该图书，请重新输入图书号。\n");

}

}

while(flag2!=1);

printf("请输入您要购买的数量：");

scanf("%d",&amount);

total=amount\*(pb->price);

if (pm->points>=total)

{

printf("您购买的图书为%s，数量为%d，总额为%d。\n",pb->name,amount,total);

printf("是否购买？（y/n 或 Y/N）：");

char ch;

scanf(" %c",&ch);

if (ch=='n'||ch=='N')

{

printf("取消购买。\n");

}

else if (ch=='y'||ch=='Y')

{

printf("购买成功！\n");

pm->points=pm->points-total;

}

else

{

printf("没有输入y/n 或 Y/N，购买失败。");

}

}

else

{

printf("您的余额不足，购买失败。\n");

}

printf("是否继续购买？（y/n 或 Y/N）:");

char ch;

scanf(" %c",&ch);

if (ch=='n'||ch=='N')

{

printf("停止购买 ,按任意键返回.");

getch();

break;

}

}

}

/\*释放内存\*/

void FereMemory(struct member \*head1,struct book \*head2)

{

struct member \*p1=head1,\*pr1=NULL;

struct book \*p2=head2,\*pr2=NULL;

while (p1!=NULL)

{

pr1=p1;

p1=p1->next;

free(pr1);

}

while (p2!=NULL)

{

pr2=p2;

p2=p2->next;

free(pr2);

}

}