# 程序设计题4：店家商品信息及订单管理系统

## 1 系统的基本功能

该系统要求编写一个综合的外卖店家信息及订单管理系统。

（1）要求能够管理若干个外卖店家的菜品信息，需要实现以下功能，实现对外卖店家信息的管理。

系统内的所有信息必须以文件的方式存储在硬盘中，例如：

商品编号，商品名称，商品销量。格式如下：

01 芝士牛排饭团 772

02 咸蛋黄卷 309

03 粗薯条 30

……

（2）要求能够管理若干个订单信息，需要实现以下功能，实现对买家订单信息的管理。

系统内的所有信息必须以文件的方式存储在硬盘中，例如：

配送日期，配送商品，商品编号，配送地点，配送数量，交易金额格式如下：

2017.6.1 芝士牛排饭团 001 南京财经大学 1 15

2017.6.15 咸蛋黄卷 002 南京农业大学 2 28

2017.6.18 粗薯条 003 南京邮电大学 1 18

……

## 2 操作要求

提供以下几个基本功能：

（1）系统内的相关信息文件由程序设计人员预先从键盘上录入，文件中的数据记录不得少于10条；

（2）设计并实现系统的相关界面，使用分级菜单提高人机交互性；

（3）可以添加/删除/修改管理对象的详细信息，并将改动反馈到数据文件中；

（4）统计分析店家商品信息：

⑴、xx店家商品查询：

1. 输入一个商品编号，查出此商品的相关信息并显示输出。
2. 输入一个商品名称，查出此商品的相关信息并显示输出。
3. 输入相应的商品排名，查出对应信息并显示输出。

⑵、订单状况查询：

1. 输入一个订单编号，查出此订单的相关信息并显示输出。
2. 输入一个商品名称，查出此订单的相关信息并显示输出。

（5）排序功能：能实现由用户选择按各项数据升序或降序排序对查询出的信息进行显示。

## 3 结构

（1）程序的总体框图如下：

添加订单信息

修改订单信息

删除订单信息

订单信息管理

添加商品信息

修改商品信息

删除商品信息

商品基本信息管理

xx店家商品及订单信息管理

维护销量信息

根据销量排名

商品销售信息统计

显示基本信息

管理员登录模块

图1店家信息管理系统的功能模块图

2）大体数据结构：

依据给定的店家信息，设计每个店家对应的结构类型定义如下：

struct Food {

int number; //商品编号

char name[20]; //商品名称

int sale; //商品销量

int rank; //名次

}

依据给定订单信息，设计每个订单信息对应的结构类型定义如下：

struct article {

long num1; //订单号

char name[20]; //商品名称

char dest[20]; //配送地点

int num2; //配送个数

float money； //金额

struct date deliver； //配送日期

}

### 4 其他

（1）在上述功能要求的基础上，为了提高成绩，可以添加一些额外的功能。

（2）变量、方法命名符合规范。

（3）注释详细：每个变量都要求有注释说明用途；方法有注释说明功能，对参数、返回值也要以注释的形式说明用途；关键的语句段要求有注释解释。

（4）程序的层次清晰，可读性强。

## 5 开发环境

开发工具可以选择VC++ 6.0或者Dev C++等C++开发工具。

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

#include <conio.h>

#define N 1000

//订单链表

typedef struct food

{

int number; //物品编号

char name[20]; //物品名称

int sale; //销量

int sum; //总数

}FOOD;

FOOD food[N];

int shuliang=0;

void menu();

//继续操作提示

void fhzjm()

{

char biaozhi[20];

printf("\n");

printf("还需要操作么？如果需要操作请输入：yes，否则请输入：no\n");

scanf("%s",biaozhi); //是否需要操作

if(strcmp(biaozhi,"yes")==0) //继续操作

{

menu();

}

else if(strcmp(biaozhi,"no")==0) //退出操作

exit(0);

else

{

printf("请输入正确的字符，谢谢！\n");

fhzjm();

}

}

//添加订单

void DengJi()

{

int rs;

int i,k=1;

system("CLS"); //清屏

printf("请输入需要输入几个商品信息：");

scanf("%d",&rs); //输入数量

for(i=shuliang;i<shuliang+rs;i++,k++)

{

printf("请输入第%d个商品的编号：",k);

scanf("%d",&food[i].number); //输入编号

printf("请输入商品的名称：");

scanf("%s",food[i].name); //输入名称

printf("请输入销量：");

scanf("%d",&food[i].sale); //输入销量

}

shuliang=shuliang+rs;

fhzjm();

}

//删除订单

void ShanChu()

{

char shanchuinfo[10];

system("CLS"); //清屏

printf("删除全部商品信息请输入\"all\"，删除指定编号的商品信息请输入\"one\"\n");

scanf("%s",shanchuinfo);

if(strcmp(shanchuinfo,"all")==0) //删除全部商品信息

{

int j;

printf("你删除的商品信息如下：\n");

printf("-----------编号-------------商品--------------销量\t\n");

for(j=0;j<shuliang;j++)

printf("----%d-------%s-------%d\t\n",food[j].number,food[j].name,food[j].sale);

shuliang=0;

printf("删除成功\n\n");

}

else if(strcmp(shanchuinfo,"one")==0) //删除指定编号的商品信息

{

struct food \*p=NULL;

int choice;

int i,j,k=0;

printf("请输入你要删除的商品的编号:");

scanf("%d",&choice);

//找到指定编号

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

{

if(choice==food[i].number)

{

k=1;j=i;break;

}

}

if(k)

{

if(shuliang==1)

{

p=&food[0];

free(p);

shuliang=0;

}

else

{

for(i=j;i<shuliang;i++)

{

food[i]=food[i+1];

}

shuliang=shuliang-1;

}

printf("删除成功\n\n");

}

else

{

printf("输入数据错误！\n");

}

}

fhzjm();

}

//浏览订单

void LiuLan()

{

int i;

system("CLS"); //清屏

if(shuliang==0) //订单信息不在系统中

{

printf("系统里面没有任何商品的信息！\n");

}

Else //订单信息存储在系统中

{

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

{

printf("第%d个商品信息为:\t\n",i+1);

printf("-----------编号-------------商品--------------销量\t\n");

printf("------------%d--------------%s--------------%d\t\n",food[i].number,food[i].name,food[i].sale);

}

}

fhzjm();

}

//查找订单

void ChaZhao()

{

int xx;

char choice,yy[20]; //名称

int i,j,k=0;

system("CLS”); //清屏

if(shuliang==0) //订单信息不在系统中

{

printf("系统里面没有任何商品的信息！\n");

fhzjm();

}

printf("三种查找方式：编号，名称，销量\n");

printf("如果按编号查找请输1\n如果按名称查找请输2\n如果按销量查找请输3\n");

printf("请输入您查找的方式：");

scanf("%s",&choice);

if(choice=='1') //按编号查找

{

printf("请输入需要查找商品的编号：");

scanf("%d",&xx); //输入需要查找商品的编号

printf("您所查找的商品的信息为：\n");

printf("----编号----名称----销量----\t\n");

//查找指定编号

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

{

if(xx==food[i].number)

{

j=i;k=1;

printf("----%d-------%s-------%d----\t\n",food[j].number,food[j].name,food[j].sale);

}

}

if(k==0)

printf("输入信息有误：\n");

}

else if(choice=='2') //按名称查找

{

printf("请输入需要查找商品的名称：\n");

scanf("%s",yy); //输入需要查找商品的名称

printf("您所查找的商品的信息为：\n");

printf("----编号----名称----销量----\t\n");

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

{

if(strcmp(yy,food[i].name)==0) //找到订单，输出其详细信息

{

j=i;k=1;

printf("----%d-------%s-------%d----\t\n",food[j].number,food[j].name,food[j].sale);

}

}

if(k==0)

printf("输入信息有误：\n");

}

else if(choice=='3') //按销量查找

{

printf("请输入需要查找商品的销量：\n"); //输入需要查找商品的销量

scanf("%d",&xx);

printf("您所查找的商品的信息为：\n");

printf("----编号----名称----销量----\t\n");

//查找指定商品的信息

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

{

if(xx==food[i].sale)

{

j=i;k=1;

printf("----%d-------%s-------%d----\t\n",food[j].number,food[j].name,food[j].sale);

}

}

if(k==0)

printf("输入信息有误：\n");

}

fhzjm();

}

//销量排序

void PaiXu()

{

struct food\*p1[N],\*\*p2,\*temp;

int i,j;

system("CLS"); //清屏

//对销量进行排序

p2=p1;

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

{

p1[i]=food+i;

}

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

{

for( j=i+1;j<shuliang;j++)

{

if((\*(p2+i))->sum<(\*(p2+j))->sum)

{temp=\*(p2+i);\*(p2+i)=\*(p2+j);\*(p2+j)=temp;}

}

}

printf("按照销量排序之后的信息为：\n");

printf("--编号--名称--销量--\t\n");

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

{

food[i].sum=food[i].sale;

printf("--%d--%s--%d--\n",(\*(p2+i))->number,(\*(p2+i))->name,(\*(p2+i))->sum);

}

fhzjm();

}

//存储订单链表到文件

void CunChu()

{

int i;

FILE \*rs;

if((rs=fopen("food.txt","w"))==NULL) //文件打开失败

{

printf("not open");

exit(0);

}

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

{

fwrite(&food[i], sizeof(food[i]), 1, rs); //写入文件

}

if(ferror(rs))

{

fclose(rs); //关闭打开文件

perror("写文件失败！\n");

return;

}

printf("存储文件成功！\n");

fclose(rs); //关闭打开文件

fhzjm();

}

//显示订单

void DaoChu()

{

struct food t;

int i=0;

FILE\* fp = fopen("food.txt", "r"); //以只读方式打开文件

shuliang=0;

if(NULL==fp) //文件打开失败

{

perror("读取文件打开失败！\n");

return;

}

memset(food,0x0,sizeof(food));

while(1)

{

fread(&t,sizeof(t),1,fp); //从文件中读取数据

if(ferror(fp))

{

fclose(fp); //关闭打开文件

perror("读文件过程失败！\n");

return;

}

if(feof(fp))

{

break;

}

food[i]=t;

i++;

}

fclose(fp); //关闭打开文件

shuliang=i;

printf("导出文件成功！\n");

fhzjm();

}

//打印主菜单

void menu()

{

int n=0;

system("CLS");

printf(" 店家商品信息管理系统\n");

printf("---------MENU--------\n");

printf(" 1.登记商品信息\n");

printf(" 2.删除商品信息\n");

printf(" 3.浏览所有已经登记的商品\n");

printf(" 4.查找\n");

printf(" 5.根据销量排序\n");

printf(" 6.存储到文件\n");

printf(" 7.从文件导出\n");

printf(" 8.退出系统\n");

a: printf(" 请选择：");

scanf("%d",&n);

switch (n) //系统调用功能

{

case 1:

DengJi();break;

case 2:

ShanChu();break;

case 3:

LiuLan();break;

case 4:

ChaZhao();break;

case 5:

PaiXu();break;

case 6:

CunChu();break;

case 7:

DaoChu();break;

case 8:

exit(0);break;

default:

{

printf("请输入1-8之间的数字，谢谢！\n");

goto a;

}

}

}

//日期

struct date

{ int year;

int month;

int day;

};

//订单链表

struct article

{ char name[20]; //物品名称

int num1; //物品编号

char dest[20]; //目的地

int num2; //物品数量

float money; //配送金额

struct date deliver; //配送日期

struct article \*next; //指向上一次添加的订单

};

struct article \*head; //存储订单信息

int Recordcount; //订单数量

//添加订单

void add()

{

system("cls"); //清屏

struct article \*p=NULL; //新建一个订单节点

p=(struct article\*)malloc(sizeof(struct article)); //为新建的订单节点分配内存空间

printf("请输入配送年份:");

scanf("%d",&p->deliver.year);

printf("请输入配送月份:");

scanf("%d",&p->deliver.month);

if((p->deliver.month>0)&&(p->deliver.month<13)) //输入月份的规范性检查

{

printf("请输入配送日期:");

scanf("%d",&p->deliver.day);

printf("请输入物品名称:");

scanf("%s",p->name);

printf("请输入物品编号:");

scanf("%s",&p->num1);

printf("请输入物品配送地:");

scanf("%s",p->dest);

printf("请输入配送个数:");

scanf("%d",&p->num2);

printf("请输入配送金额:");

scanf("%f",&p->money);

p->next=head; //将新建的订单加入订单链表

head=p; //链表头指向新添加的订单节点

Recordcount++; //订单数量加一

}

system("PAUSE");

}

//显示所有订单的详细信息

void show()

{

system("cls"); //清屏

//遍历订单链表

struct article \*p=NULL;

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

{

printf("%s\t",p->name);

printf("%d\t",p->num1);

printf("%s\t",p->dest);

printf("%d\t",p->num2);

printf("%f\n",p->money);

printf("%d\t",p->deliver.year);

printf("%d\t",p->deliver.month);

printf("%d\t",p->deliver.day);

printf(”\n”);

}

}

//查找订单

void search()

{

system("cls"); //清屏

char namesea[30]; //名称

int type; //编号

int choice;

struct article \*p=NULL;

printf("1按名称查找\n2按类型查找\n ");

printf("请输入你的选择：");

scanf("%d",&choice);

if(choice>=1&&choice<=2) //输入选择的规范性检查

{

if(choice==1)

{

printf("请输入物品名称：");

scanf("%s",namesea);

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

if(strcmp(p->name,namesea)==0) //找到订单，输出其详细信息

{

printf("%s\t",p->name);

printf("%d\t",p->num1);

printf("%s\t",p->dest);

printf("%d\t",p->num2);

printf("%f\n",p->money);

printf("%d\t",p->deliver.year);

printf("%d\t",p->deliver.month);

printf("%d\t",p->deliver.day);

printf(”\n”);

}

}

Else //按物品编号查找订单

{

printf("请输入物品编号\n ");

scanf("%d",&type);

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

if(p->num1==type) //找到订单，输出其详细信息

{

printf("%s\t",p->name);

printf("%d\t",p->num1);

printf("%s\t",p->dest);

printf("%d\t",p->num2);

printf("%f\n",p->money);

printf("%d\t",p->deliver.year);

printf("%d\t",p->deliver.month);

printf("%d\t",p->deliver.day);

printf(”\n”);

}

}

}

else exit(0);

system("PAUSE");

}

//根据物品名称和配送金额删除对应订单信息

void delete\_rec()

{

system("cls"); //清屏

char name[20];

float money;

int choice;

struct article \*p=NULL,\*q=NULL;

p=head;

q=head;

printf("请输入要删除的物品名称：");

scanf("%s",name);

printf("请输入金额：");

scanf("%f",&money);

for(;q!=NULL;q=q->next)

{

if((head->money==money)&&(strcmp(head->name,name)==0);

//要删除的订单为列表中第一个订单

{

printf("%d\t",q->deliver.year);

printf("%d\t",q->deliver.month);

printf("%d\t",q->deliver.day);

printf("%d\t",q->num1);

printf("%s\t",q->name);

printf("%s\t",q->dest);

printf("%d\t",q->num2);

printf("%.2f\t",q->money);

printf("确认删除？\n 1是\n 2否\n");

scanf("%d",&choice);

if(choice==1) //删除

{

head=q->next;

Recordcount--;

}

else break; //取消删除

}

Else //要删除的订单不是订单列表中的第一个订单

{

if((q->money==money)&&(strcmp(q->name,name)==0))

//找到对应的订单进行删除

{

printf("%d\t",q->deliver.year);

printf("%d\t",q->deliver.month);

printf("%d\t",q->deliver.day);

printf("%d\t",q->num1);

printf("%s\t",q->name);

printf("%s\t",q->dest);

printf("%d\t",q->num2);

printf("%.2f\t",q->money);

printf(”\n”);

}

printf("确认删除？\n 1是\n 2否\n");

scanf("%d",&choice);

if(choice==1) //删除

{

p->next=q->next;

Recordcount--;

}

else break; //取消删除

}

}

}

//存储订单链表到文件

void logistic()

{

char ch;

FILE \*fp;

char filename[10];

scanf("%s",filename); //写入文件头

if((fp=fopen(filename,"w"))==NULL)

{

printf("cannot open file\n");

exit(0);

}

while(ch=getchar()!='\n') //遍历订单链表，将各个订单信息逐次写入文件

fputc(ch,fp);

fclose(fp);

ch=fgetc(fp);

while(ch!=EOF)

{ putchar(ch);

ch=fgetc(fp);

}

fclose(fp);

}

//打印主菜单

void order()

{ int n,b=1,s;

struct article \*head=NULL;

void add();

void show();

void logistic();

void search();

void change();

void delete\_rec();

while(b==1)

{

system("cls"); //清屏

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

printf("欢迎使用商品配送管理系统\n");

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

printf("1:添加记录\n");

printf("2:显示记录\n");

printf("3:查询记录\n");

printf("4:删除记录\n");

printf("5:退出\n");

printf("请输入您的选择:\n");

scanf("%d", &n);

switch(n) //系统功能调用

{case 1: add();break;

case 2: show();break;

case 3: search();break;

case 4: delete\_rec();break;

case 5: exit(0);break;

default :printf("error\n");}

printf("是否返回主菜单? 0是 1否\n");

scanf("%d",&s);b=b+s;

}

}

int main()

{

char nam[N];

char password[N]; //用于存放密码

char shop[N];

char key[N]; //用于保存确认密码

int tryAgain = 0; //重试的次数

//创建管理系统

do

{

printf("----------创建用户---------\n");

printf("请输入你的用户名：");

scanf("%s", nam);

printf("请输入你的管理密码：");

scanf("%s", password);

printf("请确认你的管理密码：");

scanf("%s", key);

}

while(strcmp(password, key)); //字符串比较函数，

printf("密码设置成功！\n您的用户名为%s\n你的密码为%s\n", nam,password);

printf("按任意键开始登录系统.....\n");

getch(); //阻塞函数，可以让程序停留在此，直到我们按下键盘上的某个按键

system("cls"); //清屏

//登录系统

printf("----------登录系统---------\n");

printf("请输入用户名：");

while(scanf("%s", shop), strcmp(nam, shop))

{

tryAgain++; //累加密码重试的次数

//密码重试的次数限制在三次以内

if(tryAgain < 3)

printf("错误%d次！请重新输入：", tryAgain);

else

break; //输入密码次数大于等于三次，退出循环

}

//输出结果

system("cls");

//两种情况 ：1输入了三次错误的密码 2输入了正确的密码

if(tryAgain >= 3)

{

printf("对不起，3次密码错误，系统退出！\n");

exit(-1); //退出程序

}

else

{

printf("请输入密码：");

while(scanf("%s", key), strcmp(password, key))

{

tryAgain++; //累加密码重试的次数

//密码重试的次数限制在三次以内

if(tryAgain < 3)

printf("错误%d次！请重新输入：", tryAgain);

else

break; //输入密码次数大于等于三次，退出循环

}

//输出结果

system("cls");

//两种情况 ：1输入了三次错误的密码 2输入了正确的密码

printf("----------输出结果---------\n");

if(tryAgain >= 3)

{

printf("对不起，3次密码错误，系统退出！\n");

exit(-1); //退出程序

}

else

{

printf("欢迎你，管理员！\n");

printf("现在进入功能选择\n"); //功能选择

lop:

printf("输入1进入商品信息管理系统\n");

printf("输入2进入订单管理系统\n");

printf("输入3退出程序\n");

int GN;

scanf("%d",&GN);

switch (GN)

{

case 1:

menu();break;

case 2:

order();break;

case 3:

exit(0);break;

default:

{

printf("请输入1-3之间的数字，谢谢！\n");

goto lop;

}

}

}

}

return 0;

}