

程序设计报告

（ 2018/2019 学年 第 二 学期）

题 目：**物流系统---用户**

|  |  |
| --- | --- |
| **专 业** | **计算机科学与技术** |
| **组长 学号姓名** | **B18030322 吴雯** |
| **组员 学号姓名** | **B18030325 张耀庭** |
|  | **B18030326 陈庚辰** |
| **指 导 教 师** | **朱旻如** |
| **指 导 单 位** | **计算机学院、软件学院、网络空间安全学院** |
| **日 期** | **2019年3月29日** |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **成员分工** | | | | | | | | | |
| 组长（吴雯） | | 用户进入系统后菜单界面的设计，选择相应的数字执行对应的功能；  用户输入产品类型后查询到相关产品，下单生成订单并保存到文件中，即函数order；  用户根据自己需要的查询方式输入内容，系统查询订单并返回结果即inquiry；  修改个人信息，完成密码的修改，新密码不一致会修改失败，即函数personal；  拓展要求 | | | | | | | |
| 组员（张耀庭） | | 代码运行，需要用户登录之后方可操作（增加了一个注册系统，新用户可以注册新账号），登录成功进入菜单界面，即函数login；  用户根据自己需要的查询方式输入内容，系统查询订单并返回结果即inquiry；  产品库存文件的录入，界面的美化； | | | | | | | |
| 组员（陈庚辰） | | 代码运行，需要用户登录之后方可操作（增加了一个注册系统，新用户可以注册新账号），登录成功进入菜单界面，即函数login；  统计一段时间内的订单详情，并输出相应的参数，即函数count；  修改个人信息，完成密码的修改，新密码不一致会修改失败，即函数personal； | | | | | | | |
| **教师评价** | | | | | | | | | |
| 评价准则 | | | | 优秀 | 良好 | | 中等 | 合格 | 不合格 |
| 该课题组掌握程序设计开发的相关工程基础知识，并能够针对求解的工程问题，进行合理的分析与综合。 | | | |  |  | |  |  |  |
| 该课题组能够结合计算机软硬件资源，合理选用算法、数据结构、数据存储方式等技术手段，对求解的工程问题进行系统设计、建模及预测。 | | | |  |  | |  |  |  |
| 该课题组能够选择合适的程序设计语言与编程开发平台，对求解的工程问题进行系统模拟、仿真与预测。 | | | |  |  | |  |  |  |
| 该课题组熟练掌握调试方法与工具，对程序开发过程中出现的问题进行分析、跟踪与调试，并在开发平台上对最终作品进行充分测试。 | | | |  |  | |  |  |  |
| 该课题组具备初步的软件开发能力，完成了课题要求的各项开发任务，功能实现全面。 | | | |  |  | |  |  |  |
| 该课题组具备初步的界面设计意识。用户使用界面布局合理，交互友好，操作简便。 | | | |  |  | |  |  |  |
| 该课题组正确、完整的回答了指导教师关于课题作品的提问，反映出其对课题内容，以及相关的工程基础知识具有较好的理解和掌握。 | | | |  |  | |  |  |  |
| 该课题组具备一定的表达能力与文字处理能力，能够熟练使用文字处理软件完成课题报告一份。报告完整记录了课题组的工作及总结，内容详实，格式规范。 | | | |  |  | |  |  |  |
| 该课题组具有较好的团队协作精神，能够努力沟通解决遇到的各种开发、工程问题。 | | | |  |  | |  |  |  |
| 该课题组具备一定自学能力与探索创新意识，能够充分利用各种网络与图书馆资源自学新知识与新技能。 | | | |  |  | |  |  |  |
| 该课题组在上机过程中遵守机房规章制度，出勤与平时表现规范。 | | | |  |  | |  |  |  |
| 最终作品和报告反映出该课题组具备较好的解决工程问题的能力。 | | | |  |  | |  |  |  |
| **本次程序设计能力达成评价** | | | | | | | | | |
| 学号姓名 | **B18030322 吴雯** | | **B18030325 张耀庭** | | | **B18030322 陈庚辰** | | | |
| 贡献度 | **高** | | **较高** | | | **较高** | | | |
| 最终成绩 |  | |  | | |  | | | |
| **备注：**  贡献度：高、较高、中等、较低、低  最终成绩：优秀、良好、中等、及格、不及格 | | | | | | | | | |
| **指导教师： 朱旻如 2019 年 月 日** | | | | | | | | | |

**物流系统---用户**

**一、课题内容和要求**

物流­­---用户系统是通过登录用户的账号，选择需要的操作来进行商品的选择购买，并显示订单信息并录入系统事先建立好的订单文件中，以及根据购买时间、产品种类、产品名称进行订单查询、产品推荐、产品购买数量价格总金额的统计，若有需要可以进行个人资料（如登录密码）修改的系统。

**二、需求分析**

物流---用户系统的功能框架图如图1所示。

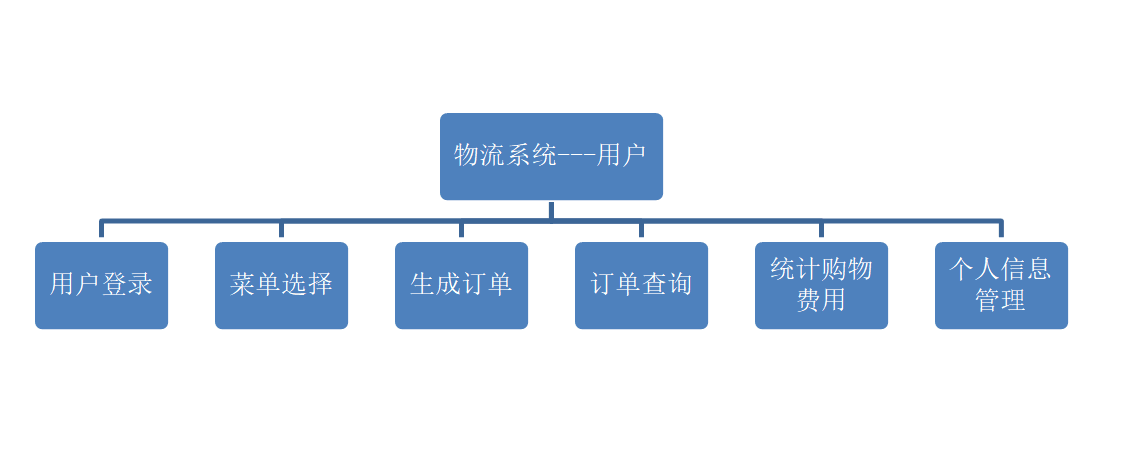


图1 功能框架图

（1）欢迎界面。

（2）提供可操作的主菜单：输出菜单，用于显示用户可以选择的功能选项。根据客户输入的选项来运行不同的功能，运行不同的函数。

（3）若选择“购买产品、生成订单”，则可以进行商品的选择和购买并存入订单文件中生成订单。

（4）若选择“查询订单、显示订单”，则可以通过输入商品种类、名称等信息进行订单查询。

（5）若选择“统计购物费用”，则可以通过输入商品名称进行订单内商品购物费用的累加并输出。

（6）若选择“个人信息管理”，则可以通过输入原密码判断正确后输入两次新密码进行密码的修改。

（7）通过对订单内一次性购买最多商品的统计向用户推荐购买商品。

**三、概要设计**

1 主要存储结构

（二级标题格式：宋体，小四号，不加粗，两端对齐，不缩进，下同）

struct Stuff //用户文件结构

{

char name[10]; //用户名

char password[20]; //用户密码

}stu; //结构体变量

struct Goods //产品信息结构

{

char name[20]; //产品名称

char type[20]; //产品类型

char cost[20]; //产品价格

char stock[20]; //产品库存

}goods; //结构体变量

char file\_str[1024]={“”}; //存储文件的每一行的字符串

char find\_str[100]={“”}; //存储用户输入的字符串

char na[10],ty[10]; //存储用户输入的名字和产品种类

char arr[100][100]; //存放分解后的字符串数组

char ptr[100]; //存放每个订单里的购买数量的字符串数组

2 主要函数流程

（1）注册登录函数：用户选择注册或登录，从文本中载入信息并判断用户名和密码是否匹配。此处用strcpy和strcat函数来进行密码匹配验证，以此来判断登录用户名和密码是否匹配；通过将用户名和密码写入保存到用户文件中进行注册。注册登录函数流程图如图3.1所示。

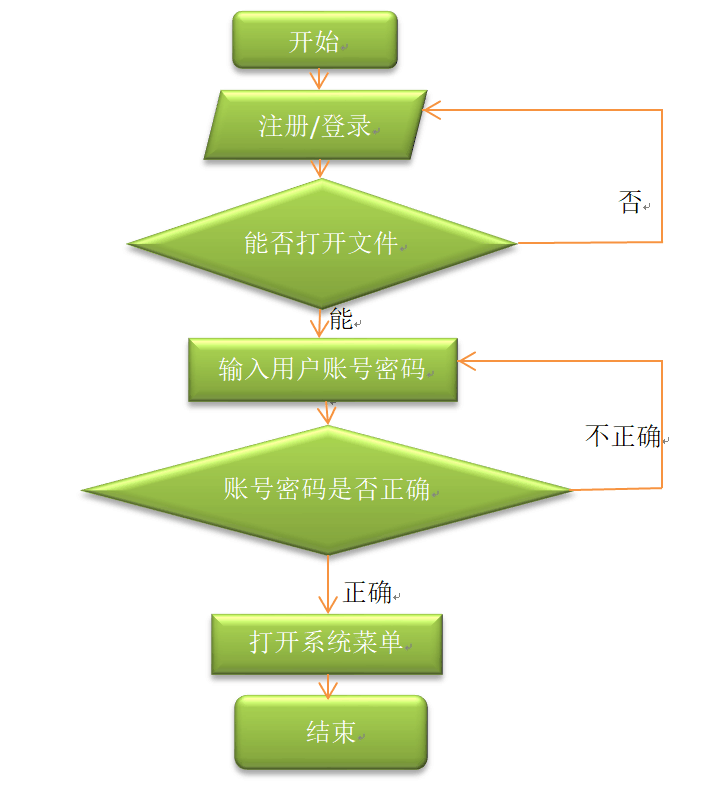


图3.1 注册登录函数

（2）订单函数：利用用户输入的产品类型到产品文件中查找并将找到的产品信息输出供用户参考，用户再输入需要购买的产品名称及数量生成订单。流程图如图3.2所示。

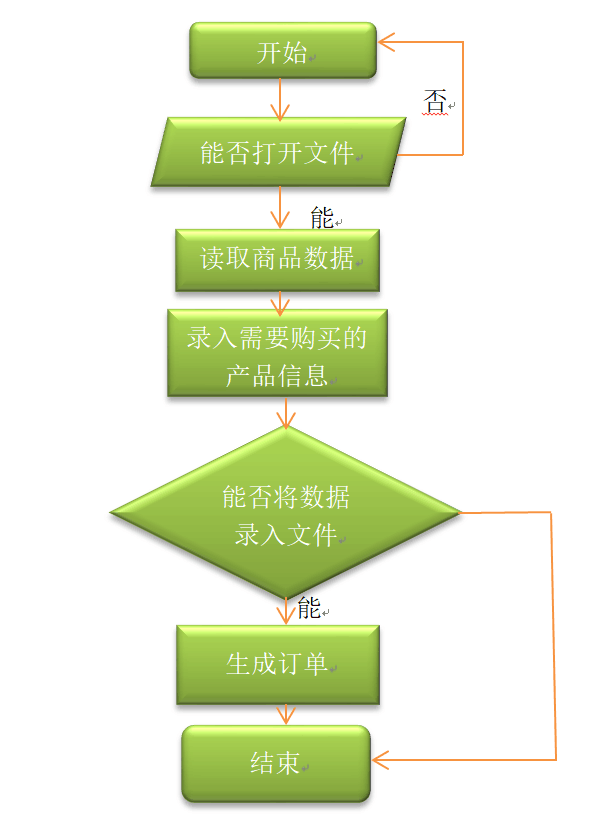


图3.2 订单函数

（3）查询函数：用户输入想查询的字符串，调用fgets函数一行一行的遍历，再用strstr函数在每行里面判断用户输入的子字符串是否包含，包含则输出；在关联查询中还用strtok函数将所需的数量分离出来相加得到购买总数。流程图如图3.3所示。

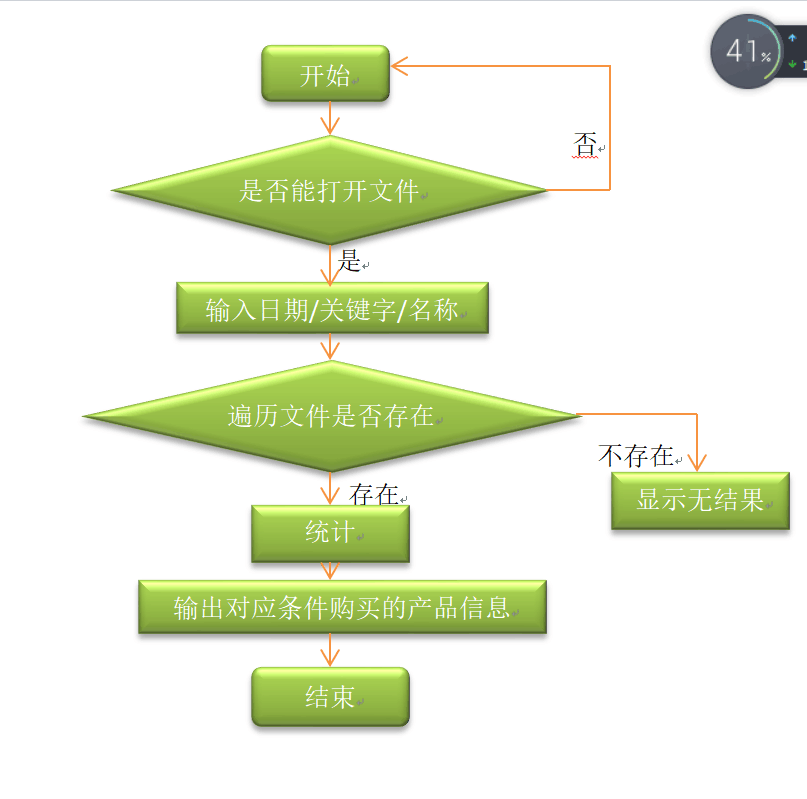


图3.3 查询函数

（4）统计函数：用户输入想要查询的产品名称，fgets遍历，strstr函数查找，strtok函数分解相加得总数并获得价格，计算得出购买金额。流程图如图3.4所示。

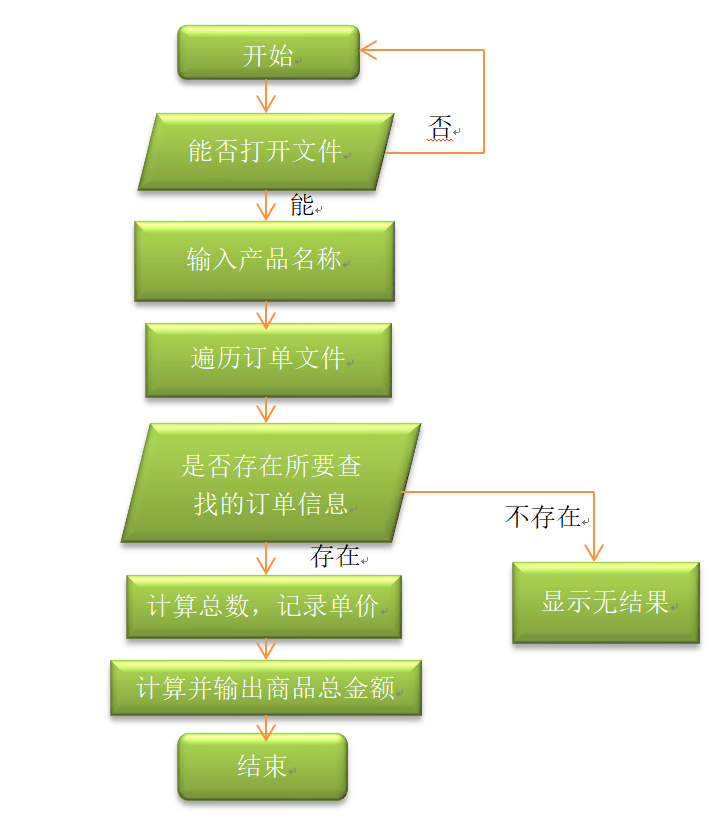


图3.4 统计函数

（5）个人信息修改函数：用户输入原始密码验证身份通过，再输入新函数，用strcmp函数判断两次新密码是否一样，一样就修改成功，并写入到用户文件中，否则修改失败。流程图如图3.5所示。

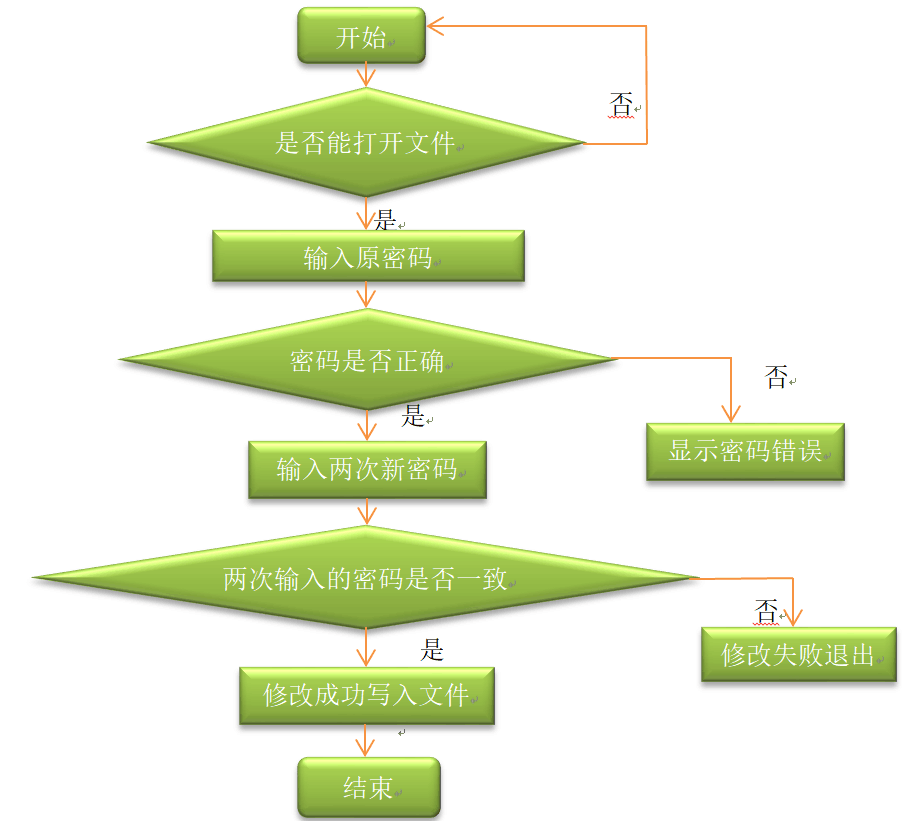


图3.5 个人信息修改函数

**四、源程序代码**

head.h

/\*头文件\*/

void read();

void login();

void order();

void inquiry();

void personal();

void count();

void expand();

login.c

/\*登录：输入用户名口令完成登录\*/

#include <stdio.h>

#include <windows.h>

void UserReg();

void Login();

struct Stuff

{

char name[10];

char password[20];

}stu;

void login()

{

int selectNum;

printf\_s("\nRegistered and Login system\n\n");

printf\_s("1 UserReg\n");

printf\_s("2 Login\n");

printf\_s("3 Exit\n");

scanf\_s("%d",&selectNum);

switch(selectNum)

{

case 1:UserReg();

break;

case 2:Login();

break;

case 3:printf("Exit Program!\n");

exit(1);

break;

}

}

void UserReg()

{

FILE \*fp;

char Linedata[50]={0},User[20],Pass[20];

fp=fopen("D:\\file1.txt","a");

printf\_s("Registered Account:\n");

printf\_s("Please input your name:\n");

fflush(stdin);

gets(stu.name);

printf("Please input your password:\n");

fflush(stdin);

gets(stu.password);

strcpy(Linedata,stu.name);

strcat(Linedata," ");

strcat(Linedata,stu.password);

strcat(Linedata,"\n");

fputs(Linedata,fp);

fclose(fp);

printf("Registered Success!\n");

}

void Login()

{

FILE \*fp;

int find=0;

char Userstrcat[50]={0};

char Userdata[50]={0};

fp=fopen("D:\\file1.txt","r");

printf("Login Account:\n");

printf("Please input Username:\n");

fflush(stdin);

gets(stu.name);

printf("Please input Password:\n");

fflush(stdin);

gets(stu.password);

strcpy(Userstrcat,stu.name);

strcat(Userstrcat," ");

strcat(Userstrcat,stu.password);

strcat(Userstrcat,"\n");

while(!feof(fp))

{

fgets(Userdata,19,fp);

if(strcmp(Userdata,Userstrcat)==0 )

{

printf("OK!Login Success!\n");

printf("Welcome back,%s!\n",stu.name);

find=1;

break;

}

}

if (!find)

{

printf("Username or Password incorrect!\n");

}

fclose(fp);

getchar();

}

order.c

/\*订单：选择产品类型显示相应产品、库存和价格，

录入需要购买的产品和数量，生成订单，

连同下单时间存入订单文件中\*/

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

const unsigned char\* COMPLICE\_DATA = \_\_DATE\_\_;

const unsigned char\* COMPLICE\_TIME = \_\_TIME\_\_;

struct Goods

{

char name[20];

char type[20];

char cost[20];

char stock[20];

};

typedef struct Goods Goods;

void order()

{

FILE \*fp;

Goods g[100],G[100];

int i=0,j=0,n,m,k;

char ty[20],na[20];

const char\*head="名称 类型 价格 购买数量 剩余库存 购买时间\n";

fp=fopen("D:\\file2.txt","r");

if(fp==0)

{

printf("file error!");

exit (1);

}

printf("请选择您所需要的产品类型(aa,bb,cc,dd):\n");

scanf("%s",ty);;

fscanf(fp,"%[^\n]s",g);

/\*读最前面一行\*/

printf("%s\n",g);

fscanf(fp,"%s%s%s%s\n",&g[i].name,&g[i].type,&g[i].cost,&g[i].stock);

while(!feof(fp))

{

i++;

fscanf(fp,"%s%s%s%s\n",&g[i].name,&g[i].type,&g[i].cost,&g[i].stock);

n=i;

}

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

{

if(strcmp(ty,g[i].type)==0)

{

printf("%s\t%s\t%s\t%s\n",g[i].name,g[i].type,g[i].cost,g[i].stock);

G[j]=g[i];

j++;

k=j;

}

}

fclose(fp);

fp=fopen("D:\\order.txt","a+");

if(fp==0)

{

printf("file error!\n");

exit (1);

}

fputs(head,fp);

printf("您所需要购买的产品和数量：\n");

scanf("%s%d",na,&m);

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

{

if(strcmp(na,G[i].name)==0)

{

printf("名称：%s 类型：%s 价格：%s 购买数量%d\n",G[i].name,G[i].type,G[i].cost,m);

fprintf(fp,"%s\t%s\t%s\t%d\t",G[i].name,G[i].type,G[i].cost,m);

}

}

printf("%s\t",COMPLICE\_DATA);

printf("%s\n",COMPLICE\_TIME);

fprintf(fp,"%s\t%s\n",COMPLICE\_DATA,COMPLICE\_TIME);

fclose(fp);

}

inquiry.c

/\*查询\*/

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

void inquiry( )

{

/\*简单simple查询：依据时间查询生成的订单\*/

FILE \*fp;

int num=0,j=0;

double cost;

char \*token;

char arr[200][200]={""};

char ptr1[200][200]={""},ptr2[200][200]={""};

int count=0;

char find\_str1[100],find\_str2[100],find\_str3[100];

char file\_str[1024];

getchar();

printf("简单查询/请输入您需要查询的日期：\n");

gets(find\_str1);

fp=fopen("D:\\order.txt","r");

if(fp==NULL)

{

printf("file error!\n");

exit(1);

}

printf("名称 类型 价格 购买数量 购买时间\n");

while(fgets(file\_str,sizeof(file\_str),fp))

{

if(strstr(file\_str,find\_str1))

printf("%s",file\_str);

}

fclose(fp);

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

/\*模糊vague查询：能依据搜索关键字显示相应产品及相应的订单信息\*/

printf("模糊查询/请输入您需要查找的关键字：\n");

gets(find\_str2);

fp=fopen("D:\\order.txt","r");

if(fp==NULL)

{

printf("file error!\n");

exit(1);

}

while(fgets(file\_str,sizeof(file\_str),fp))

{

if(strstr(file\_str,find\_str2))

printf("%s",file\_str);

}

fclose(fp);

fp=fopen("D:\\file2.txt","r");

printf("名称 类型 价格 库存\n");

while(fgets(file\_str,sizeof(file\_str),fp))

{

if(strstr(file\_str,find\_str2))

printf("%s\n",file\_str);

}

fclose(fp);

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

/\*关联查询：输入产品名称显示一段时间内的购买次数和数量以及价格等信息\*/

printf("关联查询/请输入产品名称：\n");

gets(find\_str3);

fp=fopen("D:\\order.txt","r");

if(fp==NULL)

{

printf("file error!\n");

exit(1);

}

printf("名称 购买次数 购买数量 价格：\n");

while(fgets(file\_str,sizeof(file\_str),fp))

{

int i=0;

if(strstr(file\_str,find\_str3))

{

//printf("%s",file\_str);

count++;

token=strtok(file\_str,"\t");

while(token)

{

strcpy(arr[i],token);

if(i==3)

{

strcpy(ptr1[j],arr[3]);

strcpy(ptr2[0],arr[2]);

num+=atoi(ptr1[j]);

cost=atoi(ptr2[0]);

j++;

}

i++;

token=strtok(NULL,"\t");

}

}

}

printf("%s\t%d\t%d\t%lf\n",find\_str3,count,num,cost);

fclose(fp);

}

count.c

/\*统计：统计时间段内或者某种产品的购物费用\*/

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

void count()

{

FILE \*fp;

char \*token;

char find\_str[100];

char file\_str[1024];

char arr[200][200]={""};

char ptr1[200][200]={""},ptr2[200][200]={""};

double cost;

int num=0,j=0;

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

printf("统计/请输入您所需要查询的产品名称：\n");

getchar();

gets(find\_str);

printf("名称 购物费用：\n");

fp=fopen("D:\\order.txt","r");

if(fp==NULL)

{

printf("file error!\n");

exit(1);

}

while(fgets(file\_str,sizeof(file\_str),fp))

{

int i=0;

if(strstr(file\_str,find\_str))

{

token=strtok(file\_str,"\t");

while(token)

{

strcpy(arr[i],token);

if(i==3)

{

strcpy(ptr1[j],arr[3]);

strcpy(ptr2[0],arr[2]);

num+=atoi(ptr1[j]);

cost=atoi(ptr2[0]);

j++;

}

i++;

token=strtok(NULL,"\t");

}

}

}

printf("%s\t%lf\n",find\_str,num\*cost);

}

personal.c

/\*个人信息管理：修改密码等个人资料修改\*/

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <conio.h>

struct Stuff

{

char name[10];

char password[20];

};

typedef struct Stuff Stuff;

void personal()

{

Stuff stu;

FILE \*fp;

char mima[20],mima1[20],mima2[20];

printf("请输入原始密码：\n");

getchar();

gets(mima);

fp=fopen("D:\\file1.txt","r+");

if(fp==0)

{

printf("file error!\n");

exit (1);

}

fscanf(fp,"%s",stu.name);

fscanf(fp,"%s",stu.password);

if(strcmp(mima,stu.password)==0)

{

printf("\t密码正确！\n");

printf("\t请输入新密码(八位)：\n");

gets(mima1);

printf("\t请再次输入新密码（八位）：\n");

scanf("%s",mima2);

if(strcmp(mima1,mima2)==0)

{

printf("\t密码正确！\n请牢记密码！\n");

strcpy(stu.password,mima1);

rewind(fp);

fprintf(fp,"%s %s",stu.name,stu.password);

getch();

}

else

{

printf("\t两次密码不一样！修改失败：\n");

}

}

else

{

printf("密码错误！\n");

}

fclose(fp);

}

expand.c

/\*拓展要求\*/

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

struct Goods

{

char name[20];

char type[20];

char cost[20];

char stock[20];

};

typedef struct Goods Goods;

void expand()

{

Goods g[100];

FILE \*fp;

char \*token;

int j=0,k=9,num[200]={0};

char file\_str[1024],find\_str[100]="2019";

char ptr[200][200];

char arr[200][200];

fp=fopen("D:\\order.txt","r");

if(fp==NULL)

{

printf("file error!\n");

exit(1);

}

while(fgets(file\_str,sizeof(file\_str),fp))

{

int i=0;

if(strstr(file\_str,find\_str))

{

token=strtok(file\_str,"\t");

{

strcpy(arr[i],token);

if(i==3)

{

strcpy(ptr[j],arr[3]);

num[j]=atoi(ptr[j]);

if(num[j]>k)

k=num[j];

j++;

}

i++;

token=strtok(NULL,"\t");

}

}

}

printf("您一次性购买次数最多产品的数量是:%d\n",k);

printf("为您推荐：ABC\n");

fclose(fp);

}

main.c

/\*主函数\*/

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

#include <conio.h>

#include "head.h"

int main()

{

//首页界面

int i=1,n;

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

printf(" \n");

printf(" ");printf("物流系统——用户\n");

printf(" \n");

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

getchar();

system("cls");

login();

getchar();

system("cls");

while(i)

{

printf("购买产品、生成订单·········1\n");

printf("查询订单、显示订单·········2\n");

printf("统计购物费用············3\n");

printf("个人信息管理············4\n");

printf("请输入您所需要执行的操作：");

scanf("%d",&n);

switch(n)

{

case 1:

order();

break;

case 2:

inquiry();

break;

case 3:

count();

break;

case 4:

personal();

break;

}

printf("你是否要继续操作？\n");

printf("否（0）\t是（1）\n");

scanf("%d",&i);

system("cls");

}

expand();

return 0;

}

**五、测试数据及其结果分析**

运行后，程序的界面如图5.1：

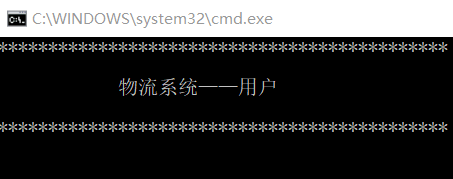


图5.1 运行后初界面

按下回车键后选择功能2进行登录，界面如图5.2：

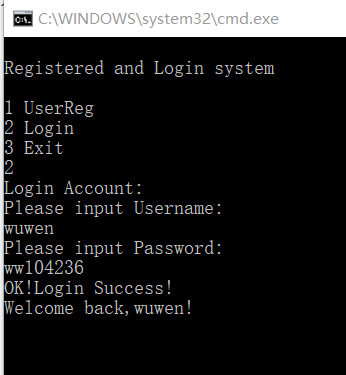


图5.2 注册/登录界面

登录成功后进入菜单界面，如图5.3：

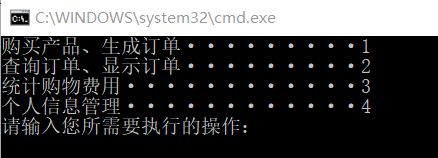


图5.3 菜单选择

输入1，执行操作1，进行购买产品，生成订单，如图5.4：

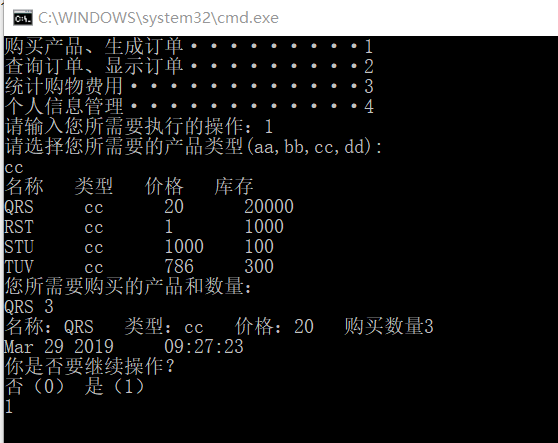


图5.4 购买产品、生成订单

继续选择操作2，执行查询功能，如图5.5：

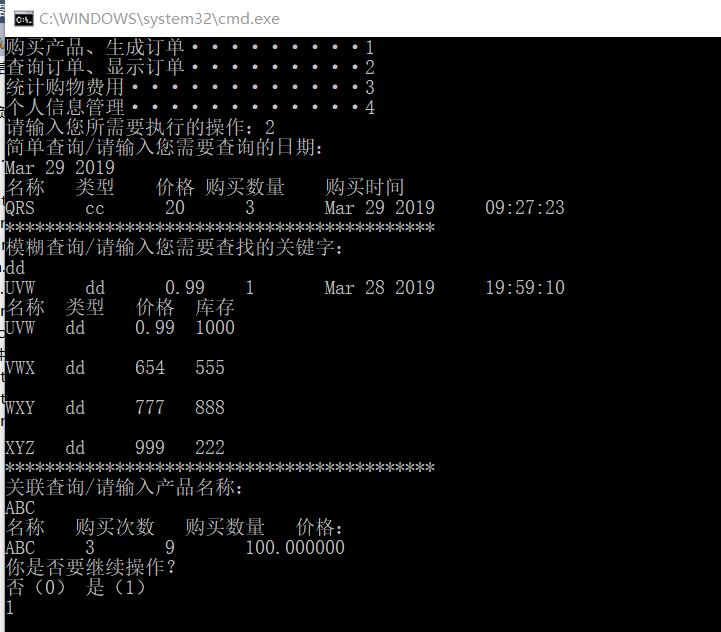


图5.5 查询订单信息

继续执行操作3，统计购物费用，如图5.6

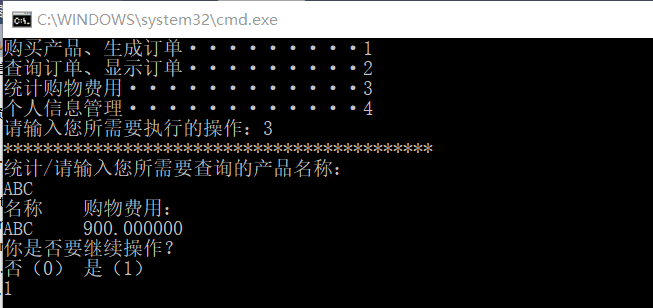


图5.6 统计购物费用

继续执行操作4，进行个人信息的修改，如图5.7：

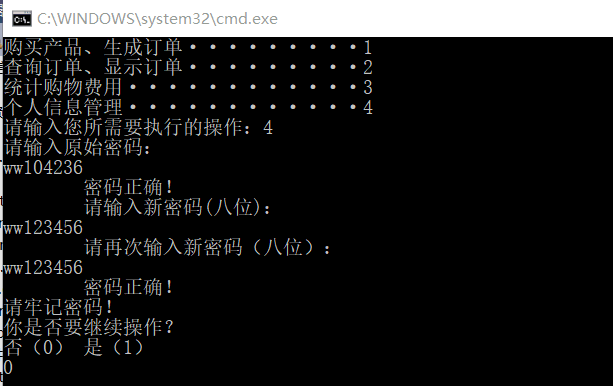


图5.7 修改个人信息

退出菜单选择功能，进入拓展要求，按照一次性购买最多的商品进行推荐，如图5.8：

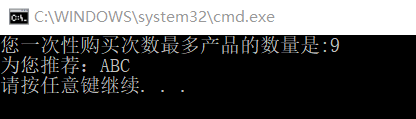


图5.8 推荐购买产品

再次按回车即退出程序。

结果分析如下。

（1）登录时如果用户名对应的密码输入错误会提示密码出错，直到重新输入正确才能进入主菜单。  
 （2）在输入菜单选项时如果输入的内容不是菜单上包含的数字，而是其他数字或者其他字符（字母或符号），系统将自动认为操作结束而推出系统。  
 （3）如果用户输入查找的产品或产品类型不在系统中，系统不会显示任何订单信息并请用户重新选择需要的操作。  
 （4）如果用户查询的内容不在文件中，则系统不会显示任何结果。  
 （5）修改密码时如果密码验证错误或者修改密码过程中两次输入的新密码不一致也无法正确修改密码，可选择退出系统或者返回主菜单。  
 （6）设置新密码时如果两次输入不同则会提示错误，可返回主菜单重新修改。

**六、调试过程中的问题**

（1）用户进行登录时出现问题，如果用户输入的用户名和用户密码不匹配，系统会给出提示：密码错误，但是继续执行下去的话还是会进入到菜单界面，无法起到保护的作用。添加一个循环语句

while(strcmp(pw,stu.password)!=0)

{

printf("wrong password!\n"); printf("     please enter the password again:    \n"); printf("                 \n");scanf("%s",pw);

}

来进行判断，密码不正确就无法登录。

（2）简单查询的过程中，输入一个需要查询的日期，查询后将所有的订单信息都显示出来，并没有起到筛选的作用，用fgets遍历文件每一行，用strstr函数判断是否包含所要的时间，包含则输出该订单

while(fgets(file\_str,sizeof(file\_str),fp))

{

if(strstr(file\_str,find\_str1))

printf("%s",file\_str);

}

（3）模糊查询中，虽然能够正常显示订单信息和产品信息，但是购买完产品后，产品库存并没有减少，还是原来的库存

fscanf(fp, "%d",atoi(stu.stock)-m);

来改变库存数目

（4）关联查询过程中，最后得出的购买总数并不是实际的购买总数，而是我定义的num的初始值，通过改变变量int i=0的位置，让其在循环体内重复初始化，达到num的多次相机，其中还用到了atoi函数把字符转换成int型

while(fgets(file\_str,sizeof(file\_str),fp))

{

int i=0;

if(strstr(file\_str,find\_str3))

{

count++;

token=strtok(file\_str,"\t");

while(token)

{

strcpy(arr[i],token);

if(i==3)

{

strcpy(ptr1[j],arr[3]);

strcpy(ptr2[0],arr[2]);

num+=atoi(ptr1[j]);

cost=atoi(ptr2[0]);

j++;

}

i++;

token=strtok(NULL,"\t");

}

}

（5）个人资料的修改，在界面中显示修改成功了，但是用户文件中的密码并没有被修改成功，只是一种浮于表面的修改，没有成功修改文件，通过在文件中读写操作，从而完成密码的彻底修改

if(strcmp(mima1,mima2)==0)

{

printf("\t密码正确！\n请牢记密码！\n");

strcpy(stu.password,mima1);

rewind(fp);

fprintf(fp,"%s %s",stu.name,stu.password);

getch();

}

（6）问题最大的应该就属于界面菜单问题了，刚开始没有菜单选择功能，整个程序就很死板，不实用，属于半成品，后来加入菜单功能供用户选择以及清屏函数system（’cls’）;还有switch语句来美化界面，提高代码的可用性

while(i)

{

printf("购买产品、生成订单·········1\n");

printf("查询订单、显示订单·········2\n");

printf("统计购物费用············3\n");

printf("个人信息管理············4\n");

printf("请输入您所需要执行的操作：");

scanf("%d",&n);

switch(n)

{

case 1:

order();

break;

case 2:

inquiry();

break;

case 3:

count();

break;

case 4:

personal();

break;

}

printf("你是否要继续操作？\n");

printf("否（0）\t是（1）\n");

scanf("%d",&i);

system("cls");

}

（7）刚开始时读取用户文件和产品文件，将其输出在屏幕上，实际使用很不安全，后来将其隐藏读取，不显示输出，停用了printf函数提高安全性，修改过后是看不到用户信息的。

**七、课程总结**

物流系统---用户是一个以文件的操作为背景的程序设计，包含了文件的打开、读写、选择、查询、修改、保存、等等操作，都是在文件的基础上操作的。我认为这个题目需要对文件有足够的了解，了解一些函数，比如strcpy，strcmp，strstr，strtok，fscanf，fprintf，fgets，atoi，等等函数来执行。个人感觉要求还是蛮高的，程序设计思路很重要，清晰的思路可以提高代码的可读性可用性，数据的处理或许可以简化处理过程，完善代码。我当时没考虑到用链表，或许用链表可以解决一部分我刚开始不能解决的问题。程序设计周，帮助我学习了之前课堂没学好的文件部分，可以说是一种课外的拓展。虽然编程过程挺烧脑，想不出来或老出bug时可能会有点失落，灰心，但是结果却很让人欣慰，有种自豪感。遇到不会的问题不要轻易放弃，现在网络发达，可以通过很多种途径获得相关知识并运用，要学会自己学习，自己克服问题。自己动手，丰衣足食！

——吴 雯

通过这次程序设计，我深切地认识了多文件工程编程的困难，尤其是在整体程序的结构设计和分工以及最后将每个人完成的代码的组合成最终的产品。尽管用的是已经学过的C语言，但依旧会犯很多低级错误以致许多bug的产生，而这些低级错误隐藏在上百行代码中，想要找到需要花费很多的精力，甚至直接从头重写一遍。而这也让我意识到尽管名义上我已经学完了C语言，但距离熟练掌握和运用还有很长的路要走。这次我们编辑的程序要用到C语言中大部分的的知识，这也使我得以知晓自己的短板，尤其是在文件的数据处理这一块，无疑是我接下来需要复习巩固加强的地方。

——陈庚辰

通过此次对物流用户系统的编写以及完善，学到了很多东西，也得出了不少宝贵的经验。首先编写此系统的代码，要求我们熟悉文件部分知识，并对其进行灵活的应用。仅仅掌握书本上的知识是远远不够的，还需要我们查阅课外资料了解各种陌生的函数并将其带入自己的程序中以实现相应的功能，从一开始对文件部分知识的不熟悉到后来能够读懂整个程序，自己在对文件的理解上提升了一个层次，整个过程通过对文件打开，编辑写入，修改，将文件中的数据信息存入数组等操作，提高了自己的逻辑思维能力，自己的动手能力也有了不小的提高，通过对程序的一步步完善，一个小组的同学之间互相学习，共同克服一个个难关，提高了我们的团队合作能力。

——张耀庭