

**课 程 设 计**

|  |  |
| --- | --- |
| **课程设计名称：** | **数据结构课程设计** |
| **专 业 班 级 ：** | **计算机科学与技术2021级2101班** |
| **学 生 姓 名 ：** | **翟向阳** |
| **学 号 ：** | **211040100127** |
| **指 导 教 师 ：** | **王峰** |
| **课程设计时间：** | **2022.12.5-2022.12.16** |

**计算机科学与技术 专业课程设计任务书**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **学生姓名** | **翟向阳** | **专业班级** | | 计科2101 | **学号** | | **211040100127** |
| **题 目** | **银行业务处理系统** | | | | | | |
| **课题性质** | 工程设计 | | **课题来源** | | | 自拟课题 | |
| **指导教师** | 王峰 | | **同组姓名** | | | 无 | |
| **主要内容** | **基本功能:**   1. 按需求完成基本的增、删、查、改、统计、排序等系统管理操作。 2. 登录功能：分为管理员登录和一般客户登录。 3. 增删查改信息功能：管理员或者客户可以对信息进行增删查改。 4. 客户注册功能；管理员通过系统让客户注册账户。   **扩展功能：**  （1）实现不同身份的登录功能，并以不同权限进行操作。  （2）用户数据以文件格式存储。  （3）客户存取款功能：客户可以进行存钱或者取钱。  （4）查看操作记录功能：管理员通过系统可以查看客户的存取款操作记录。  （5）重置系统功能：管理员可以通过重置系统，将系统内所有数据归零 | | | | | | |
| **任务要求** | （1）能够根据需求，选择合适的数据结构，确定系统的设计目标。  （2）结合软件工程的系统开发流程，利用相关开发工具，完成系统概要设计和详细设计。  （3）结合选题利用数据结构相关知识，使用C语言或C++实现该系统的所有上述功能。要求界面友善，程序运行正常。  （4）能够在设计过程中撰写规范的设计报告，在设计完成后通过答辩，并提供可运行的系统源代码电子版一套。 | | | | | | |
| **参考文献** | [1] 严蔚敏 李冬梅 吴伟民.《数据结构（C语言版）》[M]. 北京：人民邮电出版社, 2018.  [2] 谭浩强.《C语言程序设计》[M]. 北京：清华大学出版社, 2018. | | | | | | |
| **审查意见** | **指导教师签字：**  **教研室主任签字：**  **2022年 12 月 2日** | | | | | | |

说明：本表由指导教师填写，由教研室主任审核后下达给选题学生，装订在设计（论文）首页

目录

[1 需求分析 1](#_Toc122551828)

[1.1 系统概述 1](#_Toc122551829)

[1.2 系统中的角色 1](#_Toc122551830)

[1.3 功能性需求 1](#_Toc122551831)

[1.3.1 用户类型 1](#_Toc122551832)

[1.3.2 增加信息 1](#_Toc122551833)

[1.3.3 删除信息 1](#_Toc122551834)

[1.3.4 查询信息 1](#_Toc122551835)

[1.4 用例图 1](#_Toc122551836)

[2 概要设计 2](#_Toc122551837)

[2.1 抽象数据类型 2](#_Toc122551838)

[2.2 存储结构设计 4](#_Toc122551839)

[2.2.1 客户信息的存储定义 4](#_Toc122551840)

[2.2.2 收支记录的存储定义 5](#_Toc122551841)

[2.3 功能结构图 5](#_Toc122551842)

[2.4 系统子程序及功能设计 6](#_Toc122551845)

[2.4.1 乘客相关函数 6](#_Toc122551846)

[2.4.3 函数的主要关系（可选） 7](#_Toc122551847)

[函数的调用关系图，如图2-2所示： 7](#_Toc122551848)

[3 运行环境 8](#_Toc122551852)

[4 开发工具和编程语言 8](#_Toc122551853)

[5 详细设计 9](#_Toc122551854)

[5.1 MakeEmpty函数 9](#_Toc122551855)

[5.2 BianliList函数 9](#_Toc122551856)

[5.3 Load函数 10](#_Toc122551857)

[5.4 Add函数 10](#_Toc122551858)

[5.6 Delete函数 12](#_Toc122551859)

[5.7 ViewNum函数 12](#_Toc122551860)

[5.8 ViewName函数 13](#_Toc122551861)

[5.9 ViewCom函数 13](#_Toc122551862)

[5.10 SortMoney函数 14](#_Toc122551863)

[5.11 SortTime1函数 14](#_Toc122551864)

[5.12 CircleName函数 15](#_Toc122551865)

[5.13 ModName函数 16](#_Toc122551866)

[5.14 Logon函数 16](#_Toc122551867)

[5.15 Mimayincan函数 17](#_Toc122551868)

[5.16 Testl函数 18](#_Toc122551869)

[6 运行实现 18](#_Toc122551870)

[9 心得体会 26](#_Toc122551871)

[10 参考文献 27](#_Toc122551872)

**1 需求分析**

* 1. **系统概述**

本系统实现对银行的主要业务进行处理，比如开户，销户，现金的存取，查询存取记录等功能。

* 1. **系统中的角色**

|  |  |  |
| --- | --- | --- |
| 角色 | 类别 | 描述 |
| 客户 | 服务对象 | a) 更改密码；b) 查询余额；c) 浏览对账单 |
| 管理员 | 管理人员 | 增加、删除、修改、查询银行客户信息，进行现金的存取 |

* 1. **功能性需求**

**1.3.1 用户类型**

1. 管理员：可在公交线路查询系统上进行增加、删除、查询和修改的操作
2. 乘客：可在公交线路查询系统上进行查询的操作

**1.3.2 增加信息**

（1）客户开户

（2）记录客户的存取款操作记录

**1.3.3 删除信息**

（1）客户注销

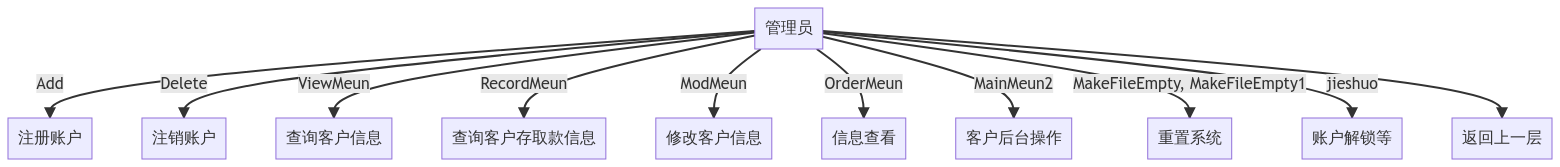
**1.3.4 查询信息**

（1）客户的信息（通过余额查看，卡号等）

（2）查询客户的操作记录（操作时间，卡号等）

* 1. **用例图**

如图1：



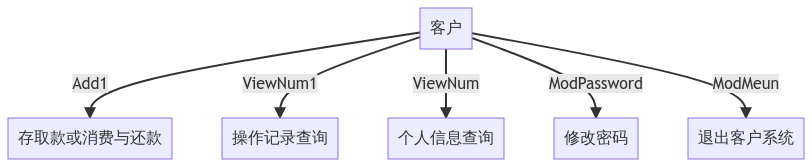


图1-1 用例图

**2 概要设计**

**2.1 抽象数据类型**

ADT Client

{  
数据对象：数据域：

char num[20]; *// 账号*

    char ID[19]; *// 身份证号*

    char name[10]; *// 姓名*

    char tel[12]; *// 电话*

    char password[20]; *// 密码*

    double money; *// 余额*

    char sd[5]; *// 锁定*

    char time[20]; *// 还款日期*

    int type; *// 卡类型1.储蓄卡，2.信用卡*

    int point; *// 信用积分*

指针域：

Position next; *// next指针 (指向下一结构体)*  
数据关系：

基本操作：

void DeleteList(List L); *// 链表删除函数(客户信息*

List MakeEmpty(List L); *// 空链表创建函数(客户信息*

void BianliList(List L); *// 遍历客户信息链表函数*

List SortMoney(List L); *// 链表冒泡排序函数(余额*

List SortNum(List L); *// 链表冒泡排序函数(账号*

void DeleteList1(List1 L); *// 链表删除函数(收支记录*

List1 MakeEmpty1(List1 L); *// 空链表创建函数(收支记录*

void BianliList1(List1 L); *// 遍历收支记录链表函数*

List1 SortNum1(List1 L); *// 操作记录(账号*

List1 SortTime1(List1 L); *// 操作记录(时间*

List Load(List L); *// 将客户信息加载到内存函数*

void Save(List L); *// 将客户信息保存到磁盘函数*

void MakeFileEmpty(); *// 清空文件函数*

List1 Load1(List1 L); *// 将收支记录信息加载到内存函数*

void Save1(List1 L); *// 将收支记录信息保存到磁盘函数*

void MakeFileEmpty1(); *// 清空文件函数*

void Choose(); *// 访客选择身份*

void MainMeun1(List L, List1 L1); *// 管理员主菜单*

void MainMeun2(char*[]*, List L, List1 L1,int flag); *// 客户主菜单*

void ViewMeun(List L, List1 L1); *// 查询客户信息菜单*

void RecordMeun(List1 L); *// 查询客户存取款记录菜单*

void OrderMeun(List L, List1 L1); *// 客户信息排序菜单*

void ModMeun(List L, List1 L1); *// 修改菜单*

void ModMeun1(Position p, List L, List1 L1); *// 修改的部分菜单*

void Add(List L); *// 信息录入函数*

void ViewNum(char num*[]*, List L); *// 账号显示函数*

void ViewName(char name*[]*, List L); *// 姓名信息显示函数*

void ViewTel(char tel*[]*, List L); *// 电话信息显示函数*

void ViewID(char ID*[]*, List L); *// 身份证显示函数*

void ViewCom(List L); *// 综合查询*

void CircleTel(char tel*[]*, Position p, List L, List1 L1); *// 电话*

void CircleName(char name*[]*, Position p, List L, List1 L1); *// 姓名*

void CircleID(char ID*[]*, Position p, List L, List1 L1); *// 身份证查询*

void ModName(Position p, List L, List1 L1); *// 客户姓名修改函数*

void ModTel(Position p, List L); *// 客户电话修改函数*

void ModPassword(Position p, List L); *// 客户密码修改函数*

void Delete(List L, List1 L1); *// 删除函数，客户信息*

void Add1(char sj*[]*, Position p, List L, List1 L1); *// 收支记录录入函数*

void ViewNum1(char num*[]*, List1 L); *// 收支记录查询函数(按账号*

void ViewName1(char name*[]*, List1 L); *// 收支记录查询函数(按姓名*

void ViewCom1(List1 L); *// 综合查询*

Position Logon(List L);

int Mimayincan(char a*[]*); *// 密码隐藏函数*

void PrintRecords(Position1 p); *// 收支记录打印函数*

void PrintClient(Position P); *// 客户信息打印函数*

int Testl(int a, int b, int c); *// 菜单输入检测函数*

void jieshuo(List L, List1 L1);

}ADT Client

**2.2 存储结构设计**

本系统中采用链表存储客户信息和客户存取款记录的信息。客户信息结构体客户

**2.2.1 客户信息的存储定义**

struct Client //

{

char num[20]; // 账号

char ID[19]; // 身份证号

char name[10]; // 姓名

char tel[12]; // 电话

char password[20]; // 密码

double money; // 余额

char sd[5]; // 锁定

char time[20]; // 还款日期

int type; // 卡类型1.储蓄卡，2.信用卡

int point; // 信用积分

Position next; // next指针 (指向下一结构体)

};

**2.2.2 收支记录的存储定义**

struct Records // 操作记录结构体

{

char num[30]; // 账号

char name[20]; // 姓名

char ID[20];

char time[20]; // 操作时间 年/月/日,时:分 2022/12/10-22.14

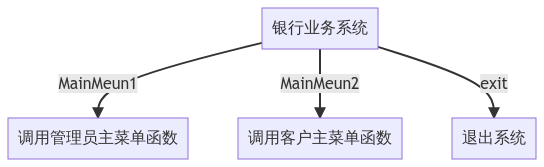
double rmb; // 存取款

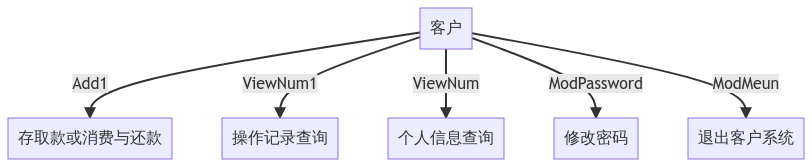
char sd[5]; // 锁定

struct Records \*next; // next指针

};

**2.3 功能结构图**

银行业务处理系统功能结构图如图2-1所示：



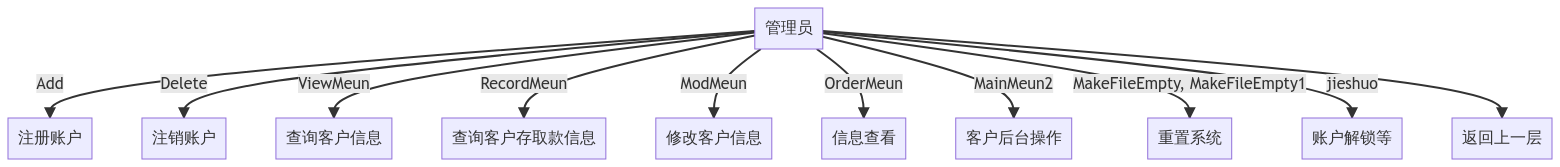


图2-1 系统结构图

**2.4 系统子程序及功能设计**

本系统的各个功能模块共涉及46个函数。

**2.4.1 乘客相关函数**

表2-1 客户相关函数

|  |  |
| --- | --- |
| 函数名称 | 功能简述 |
| DeleteList | *链表删除函数(客户信息* |
| MakeEmpty | 空链表创建函数(客户信息 |
| BianliList | 遍历客户信息链表函数 |
| SortMoney | 链表冒泡排序函数 |
| SortNum | 链表冒泡排序函数 |
| MainMeun2 | 客户主菜单 |
| ModMeun | 修改菜单 |
| ViewMeun | 查询客户信息菜单 |
| RecordMeun | 查询客户存取款记录菜单 |
| OrderMeun | 客户信息排序菜单 |
| Load | 将客户信息加载到内存函数 |
| Save | 将客户信息保存到磁盘函数 |
| MakeFileEmpty | 清空文件函数 |
| Add | 信息录入函数 |
| ViewNum | 账号显示函数 |
| ViewName | 姓名信息显示函数 |
| ViewTel | 电话信息显示函数 |
| ViewID | 身份证显示函数 |
| ViewCom | 综合查询 |
| CircleTel | 电话 |
| CircleName | 姓名 |
| CircleID | 身份证查询 |
| ModName | 客户姓名修改函数 |
| ModTel | 客户电话修改函数 |
| ModPassword | 客户密码修改函数 |
| Delete | 删除函数，客户信息 |

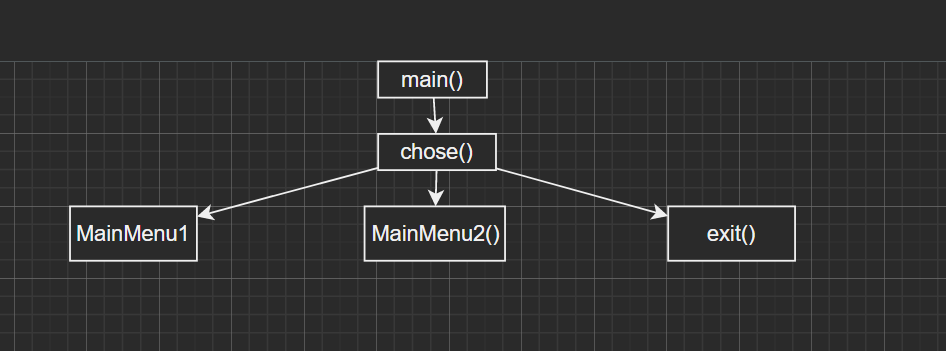
表2-1 管理员和主要功能相关函数

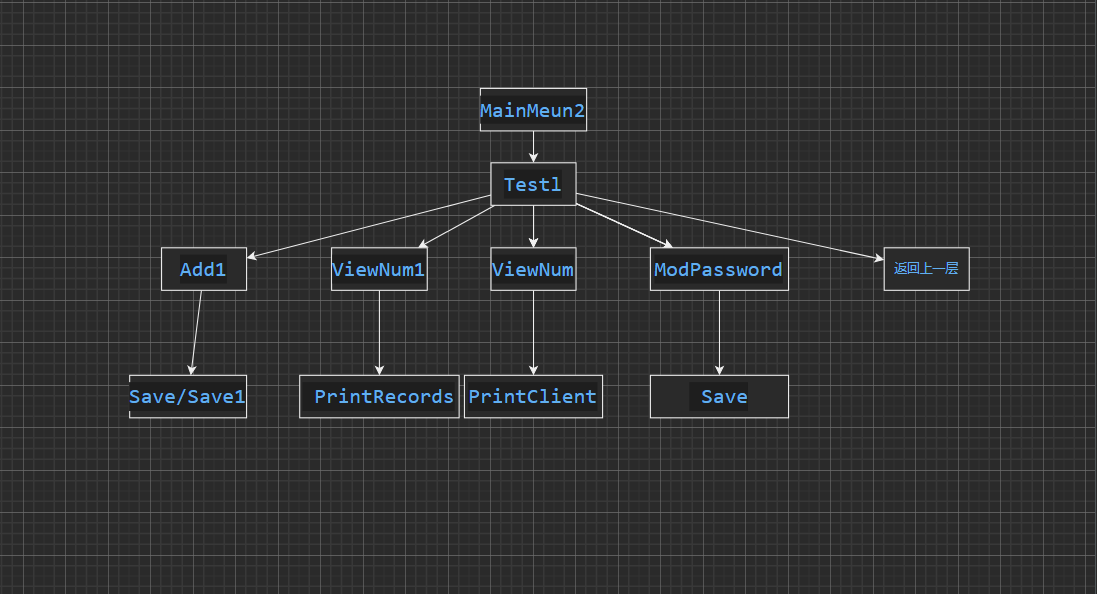
|  |  |
| --- | --- |
| 函数名称 | 功能简述 |
| DeleteList1 | 链表删除函数(收支记录 |
| MakeEmpty1 | 空链表创建函数(收支记录 |
| BianliList1 | 遍历客户信息链表函数收支记录 |
| SortNum1 | 操作记录(账号 |
| SortTime1 | 操作记录(时间 |
| MainMeun1 | 管理员主菜单 |
| ModMeun1 | 修改的部分菜单 |
| ModMeun | 修改菜单 |
| Load1 | 将收支记录信息加载到内存函数 |
| Save1 | 将收支记录信息保存到磁盘函数 |
| MakeFileEmpty1 | 清空文件函数 |
| Logon | 登录函数 |
| Mimayincan | 密码隐藏函数 |
| PrintRecords | 收支记录打印函数 |
| PrintClient | 客户信息打印函数 |
| Testl | 菜单输入检测函数 |
| jieshuo | 解锁，挂失 |
| Add1 | 收支记录录入函数 |
| ViewNum1 | 收支记录查询函数 |
| ViewName1 | 收支记录查询函数 |
| ViewCom1 | 综合查询 |

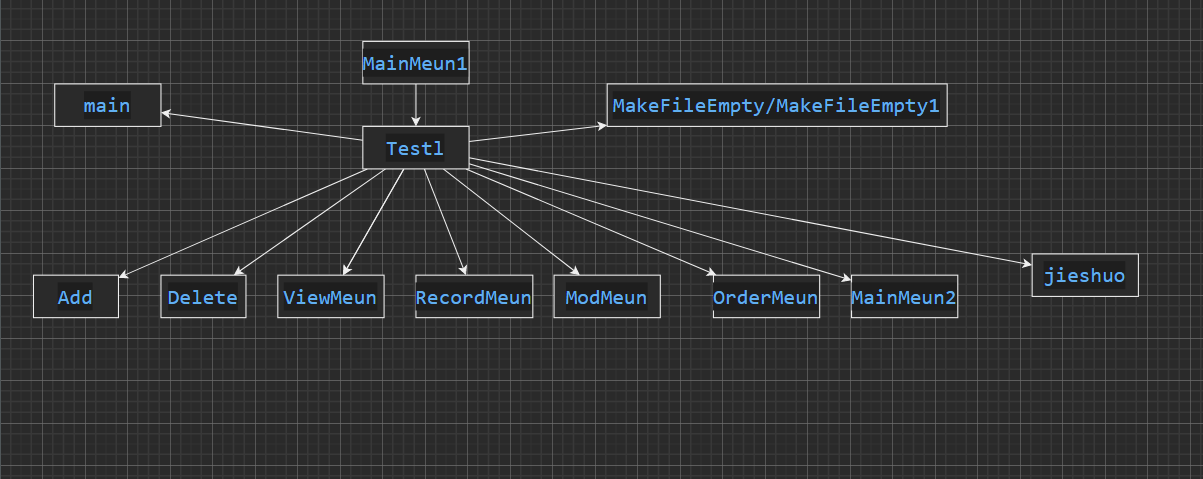
**2.4.3 函数的主要关系（可选）**

函数的调用关系图，如图2-2所示：

图2-2 函数调用关系图

****

****

****

**3 运行环境**

硬件环境： PC机内存：8G 硬盘：256G

软件环境： 操作系统：windows10

**4 开发工具和编程语言**

开发环境：VSC

编程语言：C语言

**5 详细设计**

**5.1 MakeEmpty函数**

该函数将创建一个空链表，相当于链表的初始化

if (L) // 相当于 if(L!=NULL) ,即指针L指向的空间不为空

DeleteList(L); // 删除原先链表

L = (List)malloc(sizeof(struct Client)); // 创建新的空链表

if (!L) // 相当于 if(L==NULL) ,即指针L指向的空间为空

{

printf("创建失败,内存不足!!!"); // 创建失败，提示内存不足

system("pause"); // 暂停屏幕显示

return NULL; // 返回上一级菜单

}

L->next = NULL; // Next指针指向空

return L; // 返回链表头

**5.2 BianliList函数**

遍历链表中的内容并将他们输出出来

Position p = L->next;

printf("\n客户信息如下:\n");

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

printf("账号\t\t\t姓名\t\t电话\t\t余额\t\t状态 \n");

while (p)

{

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

printf("%-20s\t%-10s\t%-11s\t%-.2lf\t\t%-5s\n", p->num, p->name, p->tel, p->money, p->sd);

p = p->next;

}

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

**5.3 Load函数**

这个是初始化函数，将文件中的客户信息内容利用头插法读入到链表中来，收支记录的读取跟此操作差不多

FILE \*fp;

Position a;

int num = 0;

fp = fopen("客户信息.txt", "r+");

if (fp == NULL)

{

printf("\n\t\t无法打开文件或文件不存在\n"); // 提示错误

printf("\n\t\t正在尝试创建新文件....\n");

fp = fopen("客户信息.txt", "w"); // 创建客户信息文件

if ((fp = fopen("客户信息.txt", "r+")) == NULL)

{

printf("\t\t文件创建失败!!!\n");

system("pause"); // 暂停屏幕显示

return NULL;

}

}

fp = fopen("客户信息.txt", "r+");//以只读的方式打开"客户信息.txt"文本文件

L = MakeEmpty(L); // 创建链表，以便存储文本文件中的信息

while (1)

{

a = (List)malloc(sizeof(struct Client)); // 创建新结点

if (a == NULL)

{

printf("\n\t\t文件信息读取失败，系统内存不足！");

system("pause"); // 暂停屏幕显示

return NULL;

}

if (fscanf(fp, "%s %s %s %s %s %lf %s %d %d %s", a->num, a->ID, a->name, a->tel, a->password, &a->money, a->time, &a->type, &a->point, a->sd) < 0)

break; // 将客户信息录入文件

a->next = L->next; // 头插法

L->next = a;

num++;

}

printf("\n\t\t客户信息读取完成，目前共有%d名客户\n", num);

fclose(fp);

return L;

**5.4 Add函数**

这是银行客户的注册功能，主要是面向过程来创建的，通过一一输入客户自身的信息来开户

Position p = L->next;

char num[20], a[7] = "正常", b[20] = "00";

int t;

printf("\n请输入所要创建的账号: "); // 提示输入账号

gets(num);

while (p != NULL && strcmp(num, p->num) != 0)

p = p->next;

if (p!=NULL)

{

printf("\n该账号已存在 ");

system("pause");

return;

}

p = (Position)malloc(sizeof(struct Client)); // 创建新结点

if (p == NULL)

{

printf("\n添加失败,内存不足!!!\n\n"); // 添加失败，提示内存不足

system("pause"); // 暂停屏幕显示

return; // 返回上一级菜单

}

strcpy(p->num, num); // 保存账号

printf("\n请输入您需要办理的卡类型(1.储蓄卡2.信用卡)： ");

scanf("%d", &t);

t = Testl(t, 1, 2);

p->type = t;

printf("\n请输入客户身份证号: "); // 保存身份证号

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

printf("\n请输入客户电话: "); // 保存电话

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

printf("\n请输入客户的姓名： "); // 保存姓名

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

printf("\n请输入客户密码： "); // 保存密码

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

p->money = 0.00; // 余额

p->point = 0; // 信用积分

strcpy(p->time, b); // 还款日期

strcpy(p->sd, a); // 状态

p->next = L->next; // 将结点p插入到链表中

L->next = p;

Save(L); // 调用 信息保存函数

system("pause");

return;

**5.6 Delete函数**

删除客户的账户，但是并不删除他的转账记录

Position p = L->next, a = L ;

Position1 p1 = L1->next, b;

char num[10];

printf("\n请输入想注销的账号: ");

gets(num); // 按账号删除

while (p != NULL && strcmp(p->num, num) != 0)

{

a = p;//记住前一个结点

p = p->next;//查找下一个结点

}

if (p == NULL)

{

printf("\n不存在该账户哦~\n");

system("pause");

return;

}

PrintClient(p); // 调用客户信息打印函数

fflush(stdin); // 清除键盘缓冲区

if (p->money > 0)

{

printf("\n请先取出账户余额！");

system("pause");

return;

}

printf("\n是否确认要删除？回复'Y'/'N'\n");

char x;

scanf("%c", &x);

if (x == 'Y')

{

Mimayincan(p->password);

a->next = p->next;

free(p);

printf("\n删除成功!!!\n\n");

Save(L); // 调用信息保存函数

Save1(L1);

}

**5.7 ViewNum函数**

通过账号检索来查询用户

Position p = L->next;

int f = 0;

while (p)

{

if (strstr(p->num, num))

{

PrintClient(p);

f = 1;

}

p = p->next;

}

if (f == 0) // 判断是否存在该客户

printf("\n没有匹配的信息");

**5.8 ViewName函数**

通过姓名检索来查询用户信息

int f = 0;

Position p = L->next;

while (p)

{

if (strstr(p->name, name))

{

PrintClient(p);

f = 1;

}

p = p->next;

}

if (f == 0) // 判断是否存在该客户

printf("\n没有匹配的信息");

system("pause");

**5.9 ViewCom函数**

通过姓名和电话号码精确搜索用户的信息

Position p = L->next;

char name[10], tel[12];

printf("输入姓名: \n");

gets(name);

printf("请输入电话: \n");

gets(tel);

int f = 0;

while (p)

{

if (strcmp(p->name, name) == 0 && strcmp(p->tel, tel) == 0)

{

PrintClient(p);

f = 1;

}

p = p->next;

}

if (f == 0) // 判断是否存在该客户

printf("\n没有匹配的信息");

system("pause");

**5.10 SortMoney函数**

根据用户的余额多少来升序排序

Position a, b, p = L->next;

int i, j, num = 0;

while (p)

{

num++;

p = p->next;

}

for (j = 0; j < num - 1; j++)

{

a = L->next;

b = a->next;

p = L;

for (i = 0; i < num - 1 - j; i++)

{

if (a->money < b->money)

{

a->next = b->next;

p->next = b;

b->next = a;

}

p = p->next;

a = p->next;

b = a->next;

}

}

**5.11 SortTime1函数**

转账记录的排序，通过用户的转账的时间大小

Position1 a, b, p = L->next;

int i, j, num = 0;

while (p)

{

num++;

p = p->next;

}

for (j = 0; j < num - 1; j++)

{

a = L->next;

b = a->next;

p = L;

for (i = 0; i < num - 1 - j; i++)

{

if (strcmp(a->time, b->time) > 0)

{

a->next = b->next;

p->next = b;

b->next = a;

}

p = p->next;

a = p->next;

b = a->next;

}

}

**5.12 CircleName函数**

修改用户的姓名

while (p) // 判断P的Next指针是否为空，且P的下一个结点的客户电话是否为查找账号

{

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

{

PrintClient(p);

printf("\n是否是您想修改的账户：回复'Y'/'N'\n");

if (getchar() == 'Y')

{

char a[20];

strcpy(a, p->password);

Mimayincan(a);

ModMeun1(p, L, L1);

}

else if (getchar() == 'N')

CircleName(name, p->next, L, L1);

}

p = p->next;

if (!p)

{

printf("\n该账号不存在： ");

system("pause");

}

}

**5.13 ModName函数**

修改用户的姓名

char name[10];

Position1 p1 = L1->next;

Position p2 = L->next;

printf("\n请输入新的姓名: ");

gets(name);

while (p2)

{

if (strcmp(p->ID, p2->ID) == 0)

{

strcpy(p2->name, name);

Save(L);

}

p2 = p2->next;

}

while (p1)

{

if (strcmp(p->ID, p1->ID) == 0)

{

strcpy(p1->name, name);

Save1(L1);

}

p1 = p1->next;

}

**5.14 Logon函数**

用户登录，该用户账户的状态正常且输入的账号密码均正确，则可以登录

char password[20], num[30], a[7] = "正常", b[7] = "锁定", c[7] = "挂失"; // 密码，账号

int f;

printf("\n\t\t请输入您的账号: ");

gets(num);

Position p = L->next;

while (p != NULL && strcmp(num, p->num) != 0)

p = p->next;

if (!p)

{

printf("\n\t\t该账号不存在！");

system("pause");

return NULL;

}

if (strcmp(p->sd, b) == 0)

{

printf("\n\t\t该账号已被锁定，请到柜台解锁！");

system("pause");

return NULL;

}

if (strcmp(p->sd, c) == 0)

{

printf("\n\t\t该账号以被挂失，请到柜台取消挂失！");

system("pause");

return NULL;

}

strcpy(password, p->password);

if (strcmp(p->sd, a) == 0)

f = Mimayincan(password);

if (f == 1)

return p;

else

{

printf("\n\t\t该账号已被锁定，请到柜台解锁！\n");

strcpy(p->sd, b);

Save(L);

system("pause");

return NULL;

}

**5.15 Mimayincan函数**

密码隐藏函数//退格不删\*

当输入的为推格键“/b”时，就会将位置往前移动一位，但是并不删除\*，重新输入密码

int i, j = 0, flag = 0;

char b[20];

int n = strlen(a);

while (j < 3)

{

printf("\n\t\t请输入您的密码(少于20位)：");

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

{

b[i] = getch();

if (i > 0 && b[i] == '\b')

{

i = i - 2;

printf("\b");

}

else

printf("\*");

}

b[i] = '\0';

j++;

if (strncmp(a, b, n) == 0)

return 1;

else if (j < 3)

printf("\n\t\t密码错误，请重新输入\n\a");

else if (flag == 0)

{

printf("\n\t\t抱歉，该账号暂时已被锁定\n");

return 0;

}

}

**5.16 Testl函数**

菜单输入检测函数

while (a < b || a > c)

{

fflush(stdin); // 清除键盘缓冲区

printf("\n\t\t\t输入错误，请重新输入: ");

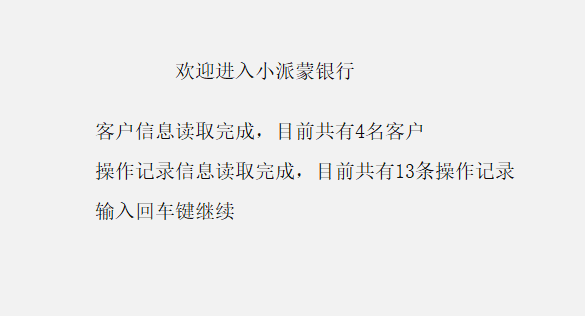
scanf("%d", &a);

}

return a;

**6 运行实现**

系统运行主界面，如图6-1所示：



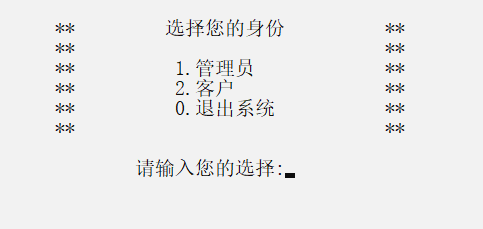


图6-1 银行业务系统主菜单

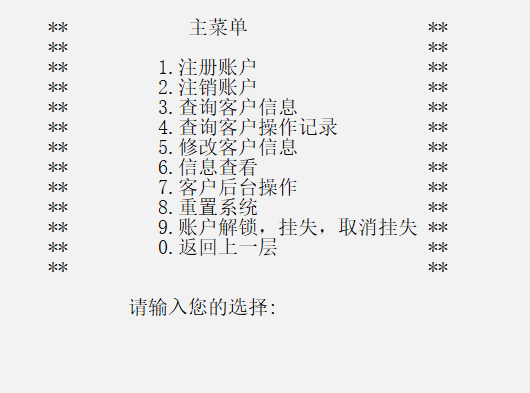


图6-2 银行业务系统管理员菜单（密码：123123）

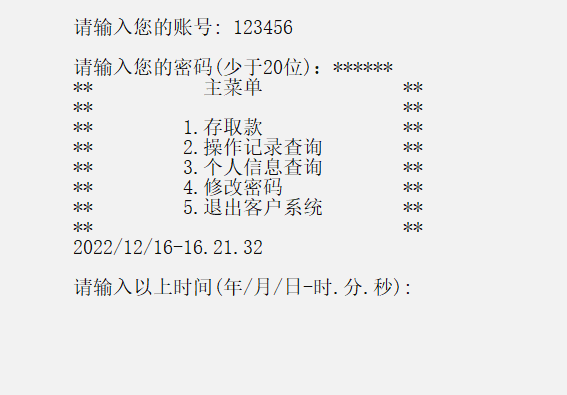


图6-3 银行业务系统客户菜单

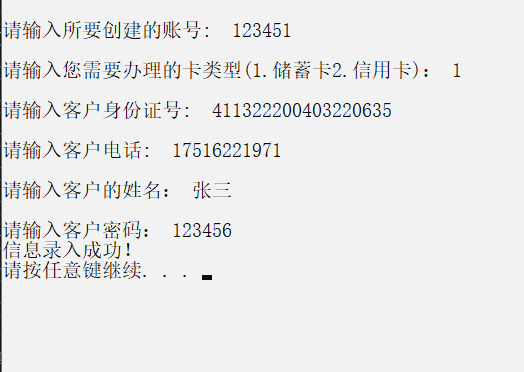


图6-4.1 银行业务系统管理员操作

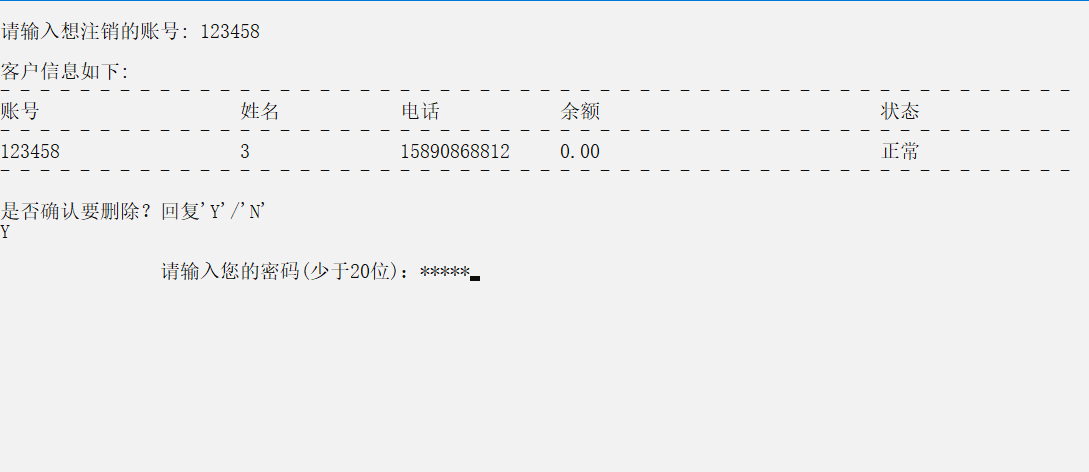


图6-4.2 银行业务系统管理员操作

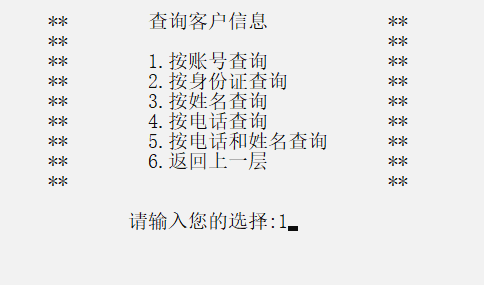


图6-4.3.0 银行业务系统管理员操作



图6-4.3.1 银行业务系统管理员操作

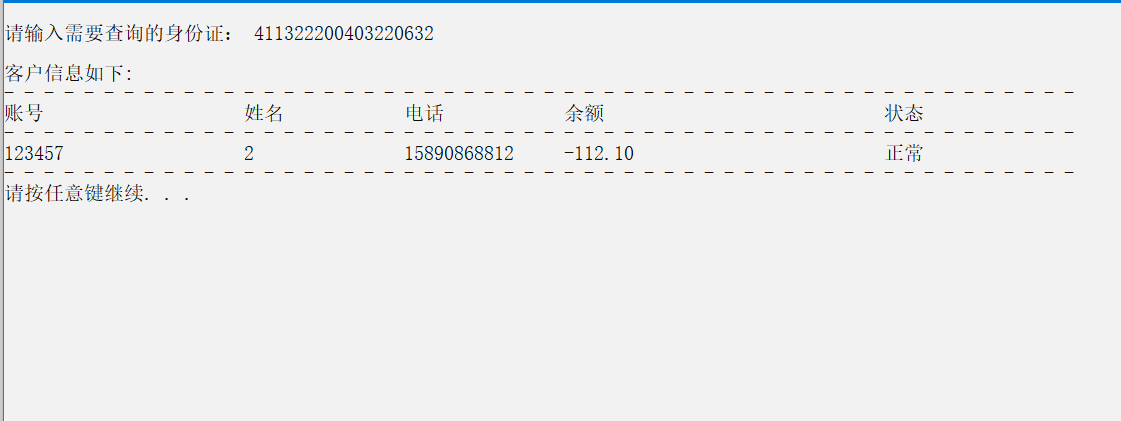


图6-4.3.2 银行业务系统管理员操作



图6-4.3.3 银行业务系统管理员操作



图6-4.3.4 银行业务系统管理员操作



图6-4.3.5 银行业务系统管理员操作

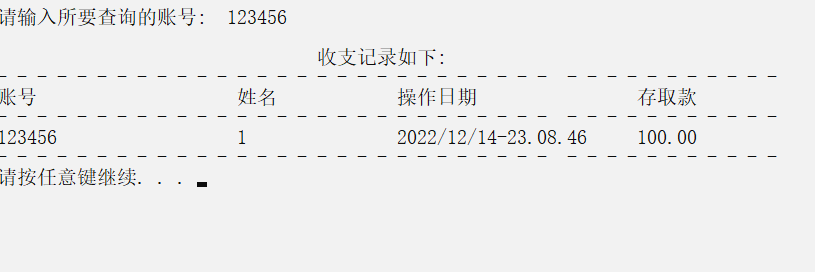


图6-4.4 银行业务系统管理员操作

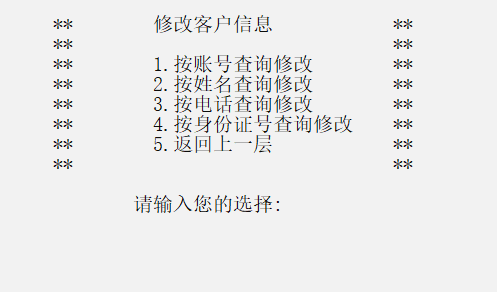




图6-4.5 银行业务系统管理员操作

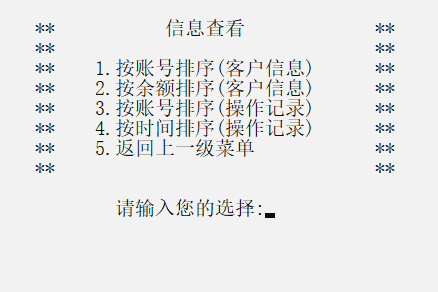




图6-4.6 银行业务系统管理员操作

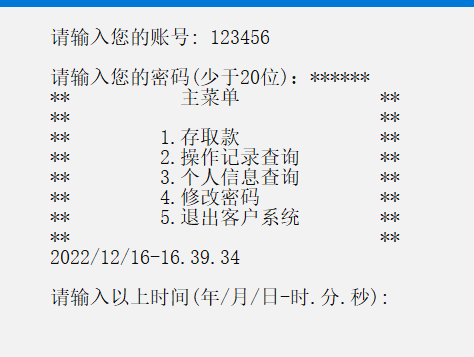


图6-4.7 银行业务系统管理员操作

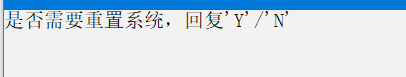


图6-4.8 银行业务系统管理员操作

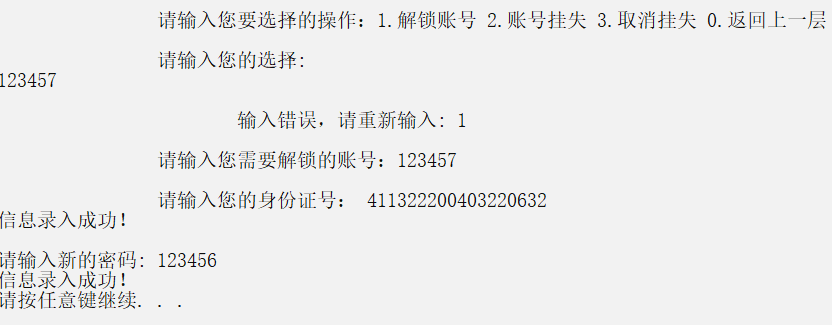


图6-4.9 银行业务系统管理员操作

客户操作：

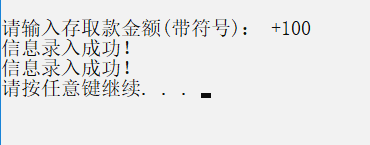


图6-4.10 银行业务系统客户操作



图6-4.11 银行业务系统客户操作



图6-4.12 银行业务系统客户操作

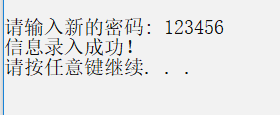


图6-4.13 银行业务系统客户操作

**9 心得体会**

本次课程设计基于上学期C++课程设计，把存储结构改成链表，并且又增加了许多功能。我对于页面的设计并不要求非常漂亮。因为我主要是为了完成系统的功能。对于一个银行系统，最基本的功能的就是要有存取款功能，在此基础上还要查询功能、转账功能，功能越多系统就越强健。

在整个过程中，我遇到了很多问题。比如自动从系统获取时间并且按照一定格式输出存储，我发现只能获取英文的，后来也去网上搜了搜，但是并没有做出来。此外，我的代码有点太长了，有一部分是功能实现的多的原因，但有些地方确实是代码重复了。

这次课程设计中，我求助了同学，不懂的问题通过询问解决了。在设计过程中我从这些同学的身上学到了很多知识，很多是以前没有听过的东西。

这次课程设计我运用的大多是老师在课堂上讲过得链表的操作。也接触到了以前没有接触到的东西。整个过程我受益良多，无论我付出了多少劳动都值得。这次课程设计也为我将来的毕业设计打下了基础。

**10 参考文献**

[1] 严蔚敏.《数据结构（C语言版）》[M]. 北京：清华大学出版社, 2018.

[2] 谭浩强.《C语言程序设计》[M]. 北京：清华大学出版社, 2018

[3]徐慧民等.C++大学基础教程[M].北京：人民邮电出版社，2005.

[4]王红梅，胡明，王涛.数据结构（C++版）.北京：清华大学出版社，2011.

[5] 苏仕华.数据结构课程设计. 机械工业出版社