

宁波大红鹰学院

信息工程学院

课 程 设 计 报 告

项目名称：家庭财务管理系统

项目组长：殷一丹

项目成员：吴思轶

班级名称：15 计科 2 班

专业名称：计算机科学与技术

完成时间：2016 年 5 月 15 日

信息工程学院制

一、案例描述

1. 总体描述

这是一款具有家庭财务管理功能的系统，所含内容包括：日期，姓名，金额，备注。

系统基本功能有：增加，修改，删除，查询，显示。附加功能包括：自动添加日期，链表的用户登录注册，找回密码，修改密码，模糊查询，批量删除，排序，导出，回收站，按时间段统计或其他统计方式等。

2. 模块描述

(1) 菜单设计：

登录界面包括：登录、注册、找回密码、修改密码。

主页面包括：收入、支出、统计、退出。

收入界面包括：1 增加 2 查询 3 修改 4 显示 5 删除 6 统计 7 排序 8 导出 9 回收站

支出界面与收入界面一致。

统计界面包括：按时间统计收入支出、按成员统计收入支出、按备注统计收入支出、统计总收入总支出、统计当前结余。

菜单界面清晰简洁，用户输入菜单对应数字即可完成相关操作。

(2) 登录模块：用户登录输入账号与密码，密码输入次数有限制，系统提醒剩余操作次数，超出限制次数，系统停止操作。当账号与密码对应一致时登录成功，进入主页面。

(3) 注册模块：用户可新建与已存在账户名不同的账户名，同时设置密码，密码输入两次，防止输入错误，另外新建用户时需要录入密保问题，以便于密码修改和密码的找回。

(4) 密码修改模块：密码修改需要原账号、密码以及正确的密保答案，即可完成密码的修改工作。

(5) 密码找回模块：密码找回需要原账号以及注册时录入的密保答案，系统即可显示密码。

(6) 增加模块：系统自动提供日期，也可录入新的日期。接着输入姓名、金额、备注。录入完一条信息后，系统弹出“是否继续添加”，用户录入 Y/y 即可继续添加，录入其他键返回菜单。

(7) 查询模块：该模块可以根据日期、姓名、金额、备注查找，还可以进行模糊查询，只要输入想要查询项的某一关键字就可以显示出含有该关键字的所有记录。

(8) 修改模块：该模块可以根据日期、姓名、金额、备注进行修改，先显示出要查找的日期（或姓名、金额、备注），系统自动为它们编号，用户选择要修改的某条信息，接着选择修改的项目，输入新的信息后即完成修改操作。

(9) 显示模块：显示该用户所有的当前信息

(10) 删除模块：该模块可以根据日期、姓名、金额、备注进行删除，先显示出要查找的日期（或姓名、金额、备注）的所有信息，系统自动为它们编号，用户选择要删除的数量和编号，即完成批量删除或个别删除操作，系统自动将所删除信息放入回收站，以便误删信息时

的找回。此外，删除包括一键删除，即清空所有信息，也可在回收站中找回。

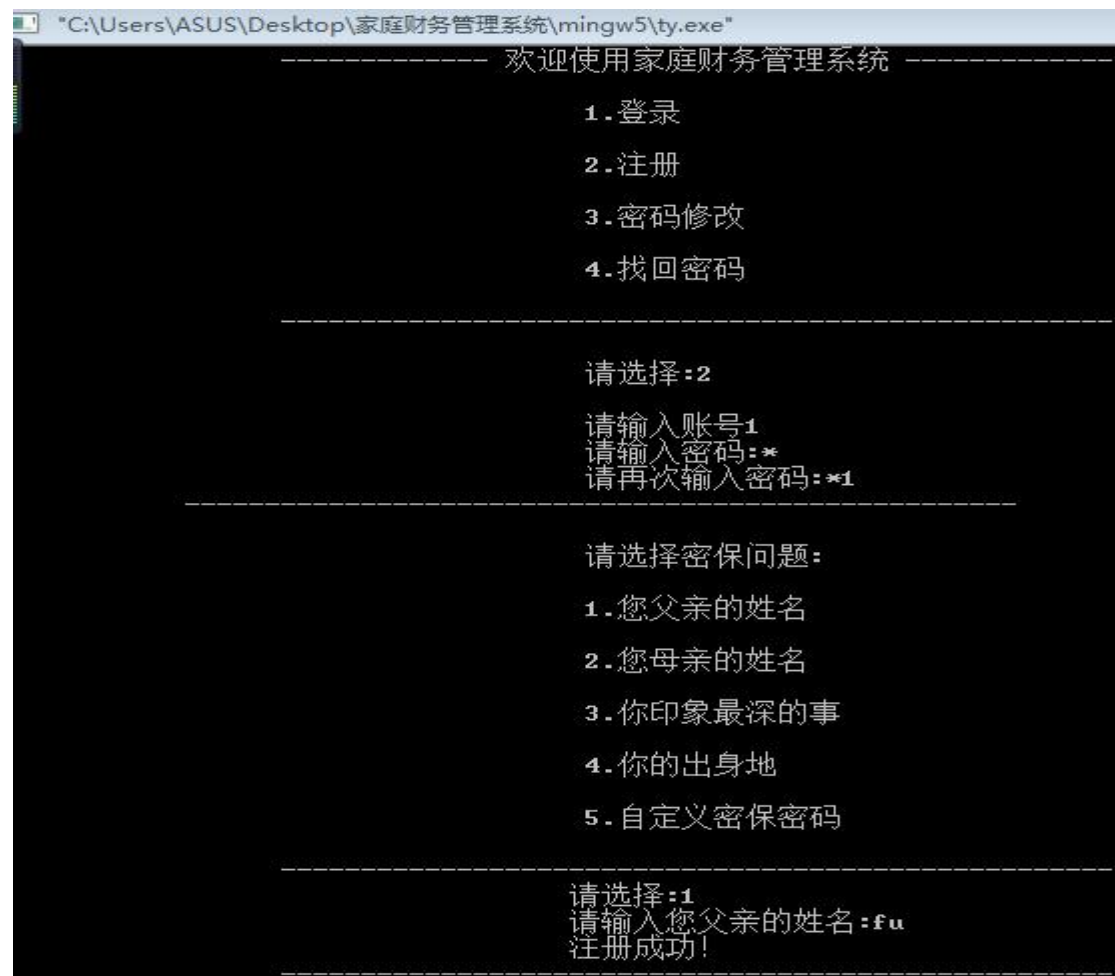
(11) 统计模块：统计分收入与支出两部分。先用 `atoi` 函数将字符型转为数值型，接着进行数据加减，按名字或备注或日期输出总支出总收入或总结余。

(12) 排序模块：包括按时间、姓名、金额、备注排序。

(13) 导出模块：以 `excel` 的形式将数据导出至桌面。

(14) 回收站模块：显示出所有用户删除的信息，系统自动为它们编号，另有恢复信息的功能，用户只需输入要恢复的数量与编号即可完成个别恢复或批量恢复的操作。

二、界面设计



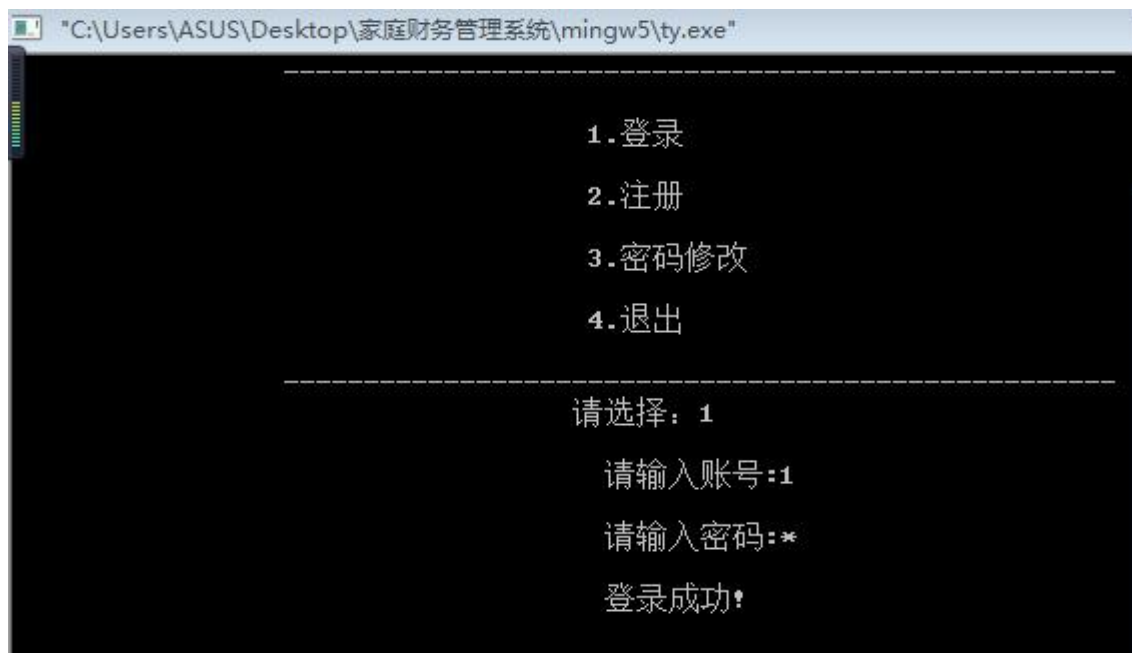


图 1 注册登录界面图

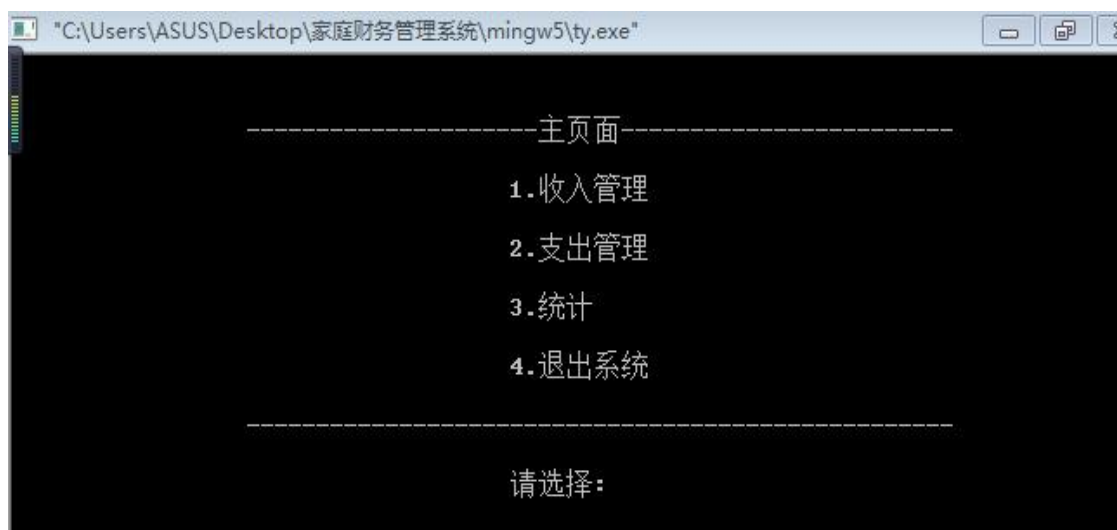


图 2 主界面图

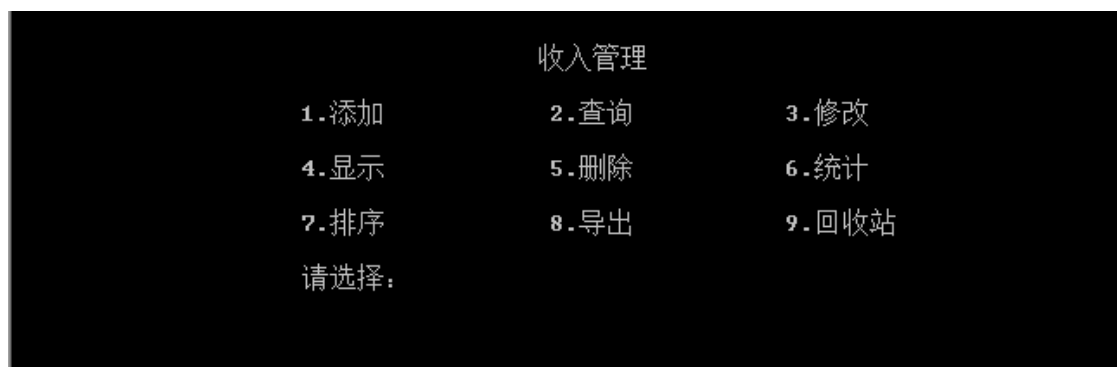




图 3 收入、支出主界面图

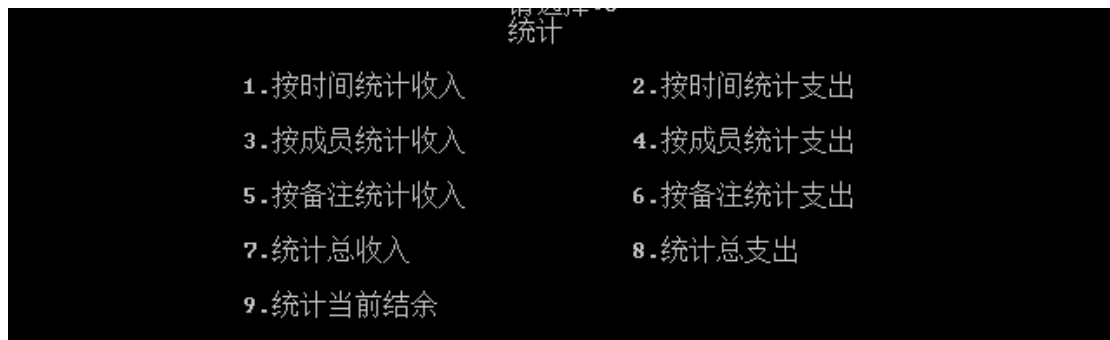


图 4 统计界面图

三、模块化设计方案

1. 系统主要函数及功能

- (1) head 函数：包括头文件、结构体以及函数的声明。
- (2) main 函数：包括全局变量的定义，链表头结点初始化，和主页面。无参数传入，无返回值，用户根据提示输入指令，用 switch 语句分析指令，进行用户登录与注册等，用缓存 (fflush(stdin);) 来保证输入有效性，用 ff 的值来控制循环。
- (3) readusername 函数：读取用户信息。
- (4) saveusername 函数：存储用户信息。
- (5) login 函数：用于用户的登录，用户有三次输入密码的机会，超出次数系统停止。密码输入时用*隐藏。
- (6) registe 函数：用于用户的注册，保证所注册的用户名不重复。同时录入了密保问题用于密码的修改与找回。
- (7) loginmodify 函数：用于修改密码，需要注册时所录入的用户名、原密码以及密保问题的答案。
- (8) getpassword 函数：用于找回密码，需要注册时所录入的用户名与密保问题的答案。
- (9) homepage 函数：主页面，用 switch 语句分析指令，进行收入操作，支出操作，统计操作。
- (10) menu 函数：子菜单用 switch 语句分析指令，进行添加、查找、修改、显示、删除、统计、排序、导出、回收站等操作。

- (11) read 函数：用于读取录入的财务信息。
- (12) save 函数：用于保存录入的财务信息。
- (13) add 函数：系统自动输出日期，用户可选择是否修改日期，接着输入其他数据，每输入一条信息后，可选择是否继续添加，添加完毕后系统自动保存数据。
- (14) search 函数：该模块可以根据日期、姓名、金额、备注查找，还可以进行模糊查询，只要输入想要查询项的某一关键字就可以显示出含有该关键字的所有记录。
- (15) update 函数：该模块可以根据日期、姓名、金额、备注进行修改，先显示出要查找的日期（或姓名、金额、备注），系统自动为它们编号，用户选择要修改的某条信息，接着选择修改的项目，输入新的信息后即完成修改操作。
- (16) show 函数：显示该用户所有的当前信息
- (17) sort 函数：包括按时间、姓名、金额、备注排序。
- (18) all 函数：统计分收入与支出两部分。先用 atoi 函数将字符型转为数值型，接着进行数据加减，按名字或备注或日期输出总支出总收入或总结余。
- (19) delet 函数该模块可以根据日期、姓名、金额、备注进行删除，先显示出要查找的日期（或姓名、金额、备注）的所有信息，系统自动为它们编号，用户选择要删除的数量和编号，即完成批量删除或个别删除操作，系统自动将所删除信息放入回收站，以便误删信息时的找回。此外，删除包括一键删除，即清空所有信息，也可在回收站中找回。
- (20) export 函数：以 excel 的形式将数据导出至桌面。
- (21) recycle 函数：显示出所有用户删除的信息，系统自动为它们编号，另有恢复信息的功能，用户只需输入要恢复的数量与编号即可完成个别恢复或批量恢复的操作。

四、数据结构描述

说明：

1. 收入支出使用了两张链表，分别表示收入支出，尾插法，尾指针其实只要遍历一下 $q=\text{head}$, $q!=\text{null}$, $q=q-\>\text{next}$, q 就是尾指针啦，但是遍历效率低。
2. 用户信息用了一张链表，这方法比较笨 多了一张链表 其实可以把用户信息放上个链表的头结点 比如 name 放用户名 detail 放密码 。
3. 其实这里 money 和 outmoney 只需要一个，一开始想的是一张链表，结果最后做成了两张链表。

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <conio.h>
#include <math.h>
```

```
#include <time.h>

struct family
{
    int year;
    int month;
    int day;
    char name[20];
    char money[20];
    char outmoney[20];
    char detail[20];
    int arrange;
};

struct list1
{
    struct family info1;
    struct list1 *next; //结构体嵌套，数据存储时无需存地址，只需存储数据。
};

struct userinfo
{
    char username[10];
    char password[10];
    char security[10];
};

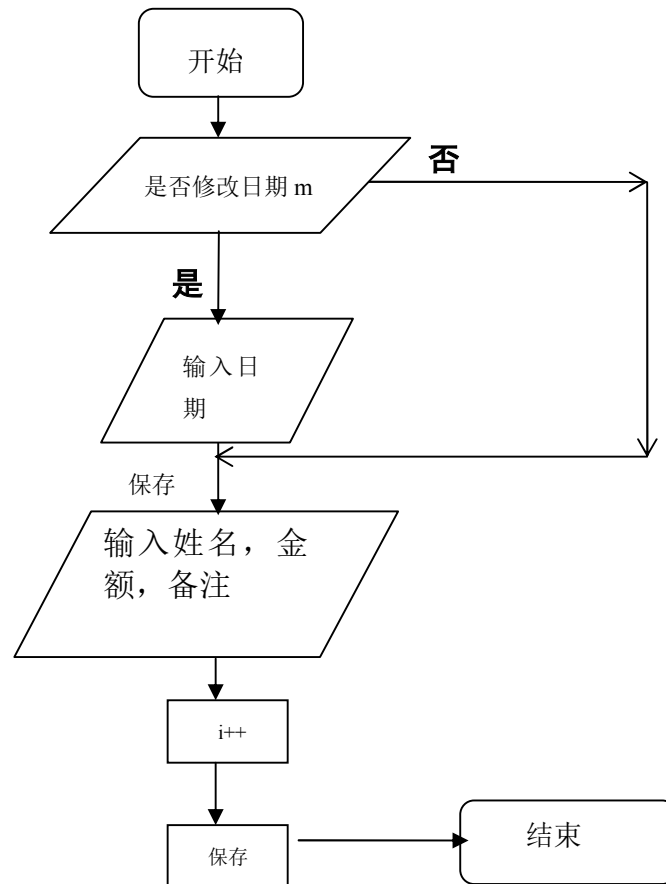
struct list2
{
    struct userinfo info2;
    struct list2 *next;
};

void readusername();
void saveusername();
```

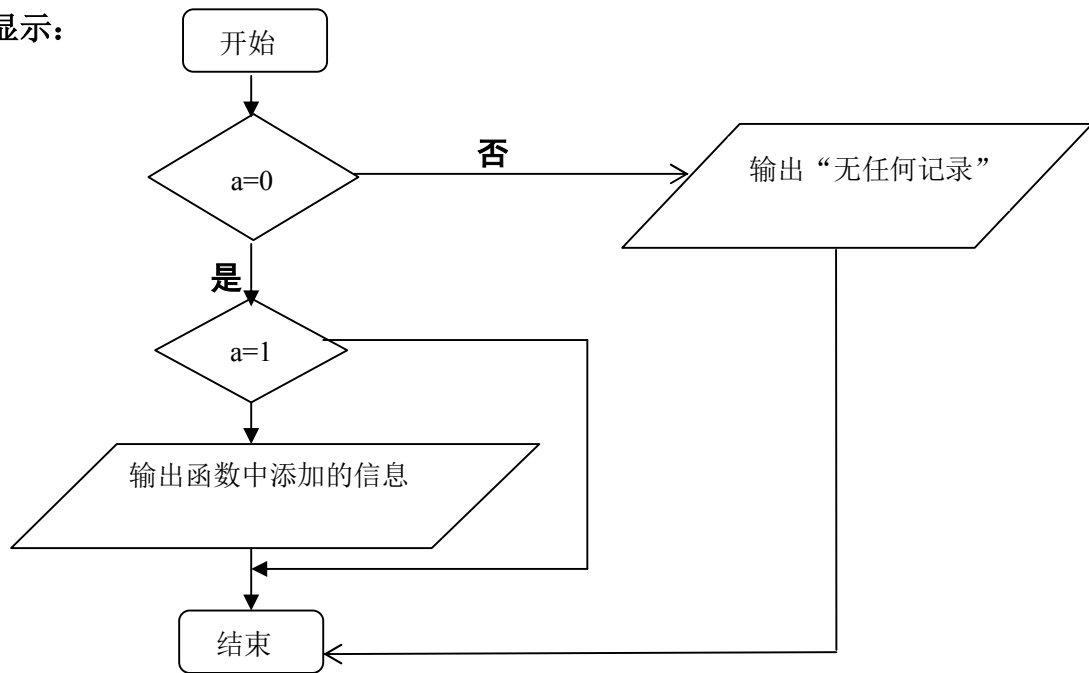
```
void login();
void registe();
void loginmodify();
void getpassword();
void homepage();
void menu(int i);//有参无返回值。i=1 进行收入操作，i=-1 进行支出操作。
void read(int i);//（下同）
void save(int i);
void add(int i);
void search(int i);
void update(int i);
void show(int i);
void delet(int i);
float all(int i);
void sort(int i);
void export(int i);
void recycle(int i);
```


五、算法设计流程

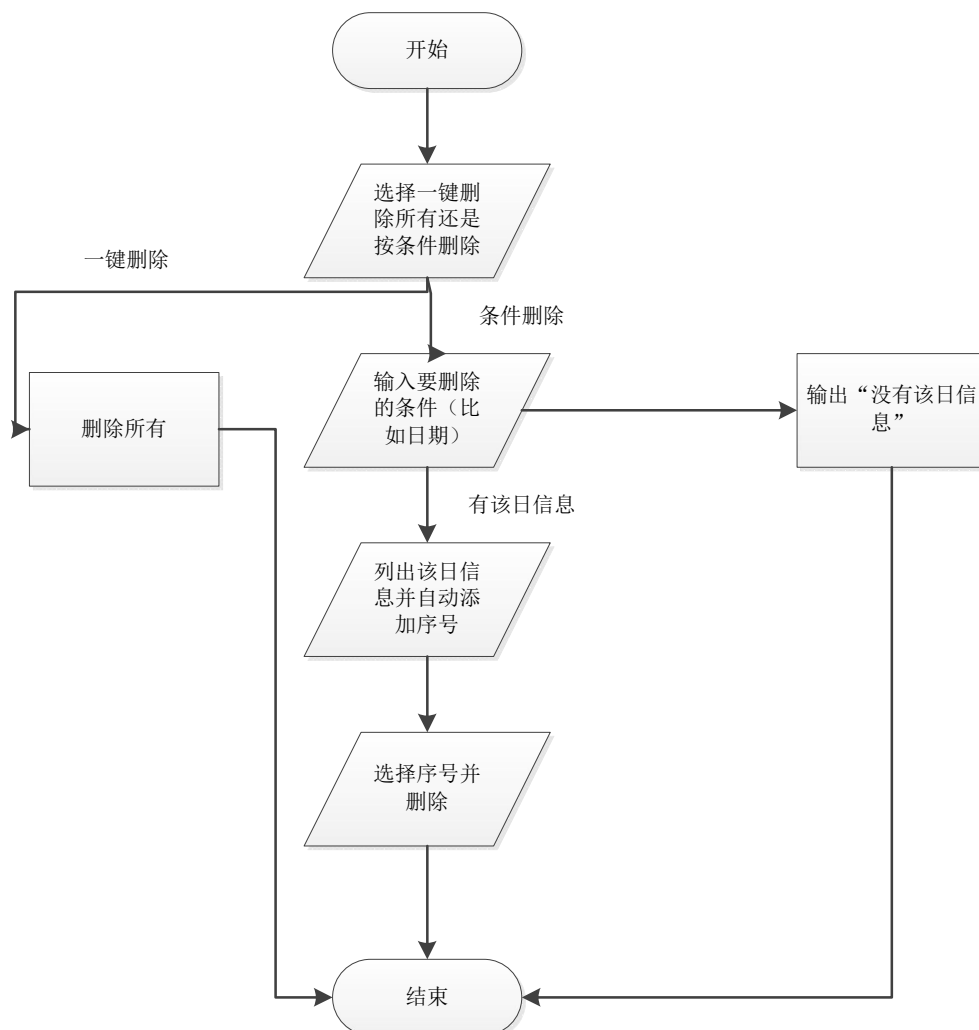
[1]添加：



[2]显示:



[3]删除:



六、程序运行结果

[1] 登录记录:

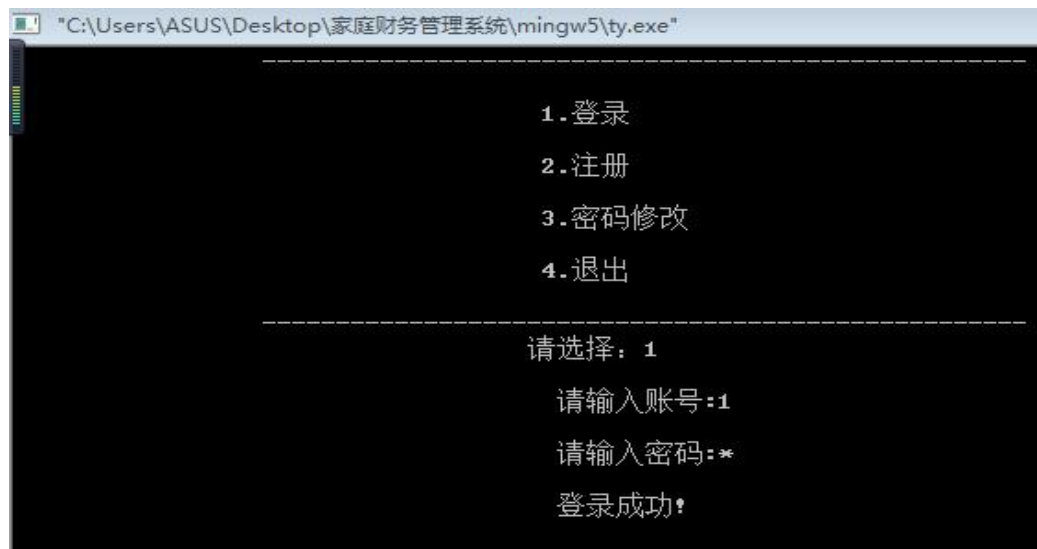


图 1 登录记录图

[2] 注册记录:

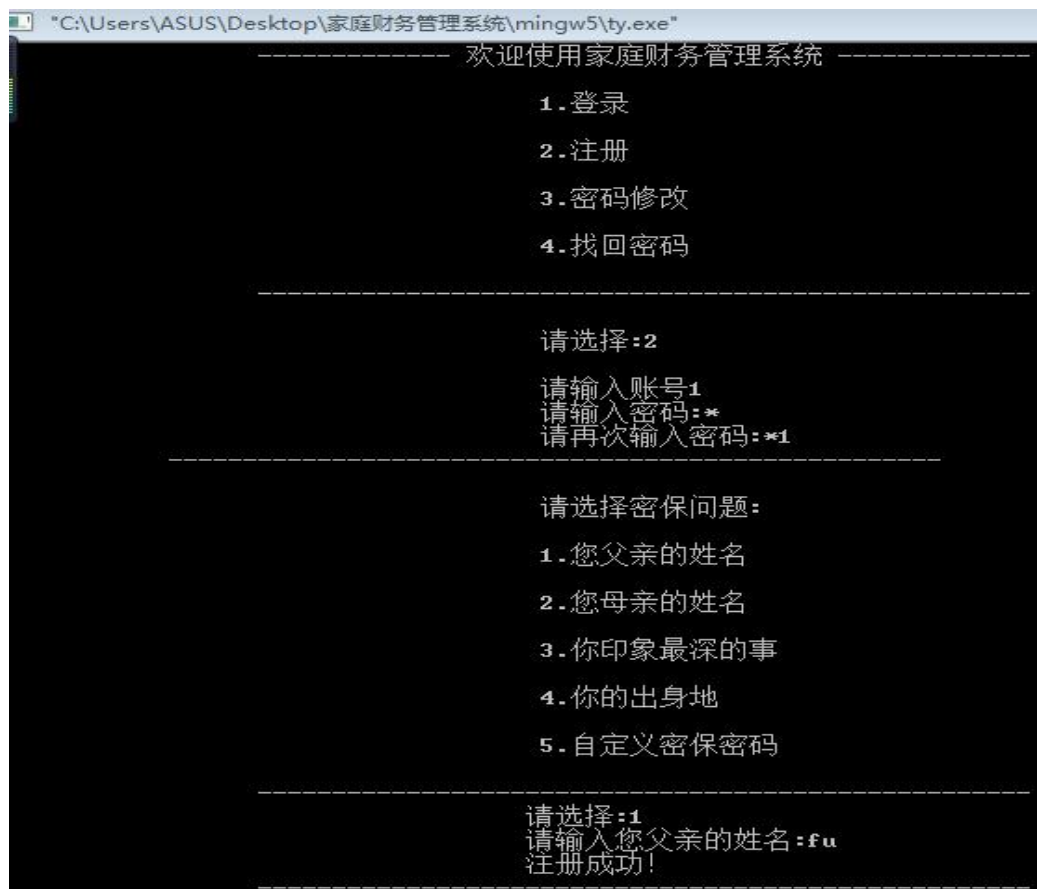


图 2 注册记录图

[3]修改密码:



图 3 修改密码记录图

[4]找回密码:



图 4 找回密码记录图

[5]添加记录:

收入管理

1.添加

2.查询

3.修改

4.显示

5.删除

6.统计

7.排序

8.导出

9.回收站

请选择: 1

日期: 2016-5-8

若修改日期 'y'/'Y'

1

请输入家庭成员姓名

ayi

请输入收入金额

100

请输入备注

1

若继续添加 'y'/'Y'

图 5 添加记录图

[6]查询记录:

1.按日期查询

2.按姓名查询

3.按金额查询

4.按备注查询

5.模糊查找

请选择:5

请输入查询关键字: 1

!NO.	!年-月-日	!类型	!姓名	!金额	!备注!
!1	!2016-5-8	!收入	!ayi	!100	!1!
!NO.	!年-月-日	!类型	!姓名	!金额	!备注!
!2	!2016-5-8	!收入	!殷一丹	!200	!2!
!NO.	!年-月-日	!类型	!姓名	!金额	!备注!
!3	!2016-5-8	!收入	!殷一旦	!200	!4!

图 6 查询记录图

[7]修改记录:

```
-----
1.按日期修改
2.按姓名修改
3.按金额修改
4.按备注修改
-----
请选择:4
请输入备注:4

!NO.      !年-月-日      !类型  !姓名  !金额  !备注!
!1        !2016-5-8      !收入  !殷一旦 !200    !4!
!2        !2016-5-8      !收入  !ayi    !400    !4!
          请输入要修改的编号:
```

图 7 修改记录图

[8]显示记录:

!NO.	!年-月-日	!类型	!姓名	!金额	!备注!
!1	!2016-5-8	!收入	!ayi	!100	!1!
!2	!2016-5-8	!收入	!殷一丹	!200	!2!
!3	!2016-5-8	!收入	!吴思轶	!300	!3!
!4	!2016-5-8	!收入	!殷一旦	!200	!4!
!5	!2016-5-8	!收入	!ayi	!400	!4!
!6	!2008-3-30	!收入	!隐隐	!500	!5!

图 8 显示记录图

[9]删除记录:

```
-----
1.按日期删除
2.按姓名删除
3.按金额删除
4.按备注删除
5.一键删除所有记录
-----
请选择:2
请输入要删除信息的姓名:ayi
INO.   年-月-日   类型   姓名   金额   备注
1      2016-5-8   收入   layi   100    11
2      2016-5-8   收入   layi   400    14
请输入要删除的数量:
```

图 9 删除记录图

[10]排序记录:

```
-----
1.按日期排序
2.按姓名排序
3.按金额排序
4.按备注排序
-----
请选择:1
INO.   年-月-日   类型   姓名   金额   备注
1      2008-3-30   收入   隐隐   500    15
2      2016-5-8    收入   layi   100    11
3      2016-5-8    收入   殷一丹 200    12
4      2016-5-8    收入   吴思轶 300    13
5      2016-5-8    收入   殷一旦 200    14
6      2016-5-8    收入   layi   400    14
若继续排序'y'/'Y'
```

图 10 排序记录图

[11]统计记录:

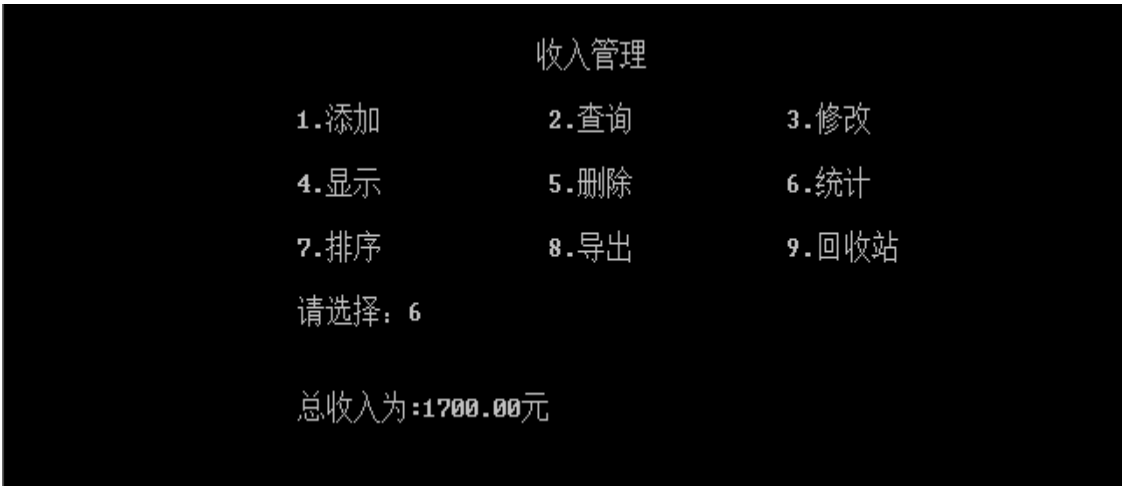


图 11 统计记录图

[12]导出记录:

	A		B		C	D	
1	年	月	日	姓名	金额	备注	
2	2008	3	30	隐隐	500	5	
3	2016	5	8	ayi	100	1	
4	2016	5	8	殷一丹	200	2	
5	2016	5	8	吴思轶	300	3	
6	2016	5	8	殷一旦	200	4	
7	2016	5	8	ayi	400	4	
8							
9							

图 12 导出记录图

[13]回收站记录:

!NO.	!年-月-日	!类型	!姓名	!金额	!备注!
!1	!2008-3-30	!收入	!隐隐	!500	!5!

图 13 回收站记录图

七、总结

1. 工作时间

第一个星期：链表基础知识掌握，基本功能完成。（把链表真正看懂用了一周）

第二个星期：附加功能。（排序用了整整一天，批量删除用了一下午，按时间段统计用了一上午，其他都是零碎时间）

第三个星期：测试 bug，又看了一下 kmp 算法，复习了一下指针，重点是指针数组（不过后来批量删除选用了更简单的方法）和数组指针

2. 心得体会

殷一丹：家庭财务管理系统，这个项目和通讯录相似度很高，但因为用的是链式存储，刚开始理解起来真的有点困难。这大概和自己是电脑小白也有点关系，不理解开辟内存，甚至啥是内存都不知道，就不要说怎么存了。第一节课上的一头雾水，于是去图书馆借了本数据结构，第一遍当然没懂，书上好多好多图，可就是看不懂啊……好吧，只能自己边画边想，满脑子想着为啥要有 next，为啥要指向，为啥要开辟内存空间，画着画着……好像慢慢的懂了！不过后来讲到删除我才发现，其实我没搞懂，所以又开始看数据结构。我知道自己有点懂的时候是在看删除的代码的时候。我奇怪于为什么有时候要考虑头节点为空，有时候不用考虑。但是我真的想不出来，所以跑去问了老师，她点拨了一句，我恍然大悟了，这才觉得自己明白了链表。可是在实际写代码的时候，还是遇到了很多困难，比如排序，指针多了难免有点烦，不过自己还是静下心来了，虽然我用了整整一天才明白，但这种豁然开朗的心情太痛快了。从系统调用时间是百度的，还是不太理解，但是这次也知道了要学会利用信息时代的资源并把它变成自己的东西。另外，自己加了一个回收站的功能，其实就是文件读取保存和删除的结合，但是同学听到感觉挺牛的。原来知识转化成项目的功能，感觉如此之好。但是真正想去实现一个功能，用到某种算法，遇到的问题实在太多了。课上老师讲到指针域不需要存，排序能否不统计次数，采用工程，设断点测试比较方便本地时间和世界时间，邻接表，字符串匹配问题，只存数据域，交换数据变得异常简单，这一个个小小的问题，深入研究起来要花一整个下午的时间甚至都不够。之前一直觉得自己特别笨，要花比别人更多的时间才可以消化，但是现在觉得，花下去的时间都是有用的，一点点的经验，很多错过的点都不会再错，都是一笔财富，希望下一个项目自己能保持热情与积极性，给自己点个赞！

吴思轶：这是我们的第二个项目，我们一人写了一个，不过很不幸的是我的被覆盖了。在这个项目中，我发现接触一门新的知识需要用心的去体会、理解。就如这个项目中的链表，我先听了老师讲的，实际上老师也没有讲多少。主要是看书，看视频，网上找资料，去把这个概念弄懂后，才能着手开始做项目。做项目才是真正把所学为所用的时候，可以说只有你做一个大点的项目出来才能真正说明你学到了东西，你会用所学的东西，要不然就算你学的再多，不会用也没用。做学问特别是计算机一定要做的精准，比如说一个链表，你一定要把它几乎所有的用方法都能弄清楚明白。遇到不会写的部分，适当的去参考借鉴他人的程序，借鉴程序并不是说一直没有目的去抄，去 copy 他人的成果。你会发现借鉴一段时间以后就不用再借鉴了，因为大部分都是一样的，一看就知道怎么写了，当一看到程序时就知道它是什么功能，哪里有错，这并不需要太多的时间，只要用心就足够了。

掌握了语言法规则，不一定就能编出程序来，其实学习编程就是要学习一种思想，一种思维方法，首先要确定一种算法，如从 1 加到 100，判断闰年等，我认为这些算法都是一种思维方法。画流程图就是在构建编写程序思路的过程，当画好一个流程图时编程思想也就出来了，程序也差不多了，再结合掌握的语言的方法规则，在计算机里实现即可了。任何程序只要有思维，有想法，那么实现程序时就不会觉得无从下手。而要学习这种思维方法就是多做题，多读程序，学习别人好的思维方法，多想想一个程序还可不可以有另外一种方法来实现它。这样久而久之，自然会形成一种自己的思维方法。因此我认为不管学什么语言都是这样。

附页（源代码附加注释）

1. main 函数

```
#include "head.h"

int h;          //全局变量 h 为信息的条数

char s[20];     //全局变量 s 为用户的账号 同时也是一对一的文件名

struct list1 *head=NULL;

struct list1 *rear=NULL;

struct list1 *outhead=NULL;

struct list1 *outrear=NULL;

//使用了两张链表 分别表示收入支出

struct list2 *usehead=NULL;

struct list2 *userear=NULL;

//账户登录的链表

int main()

{

    int num, ff=1; //ff 用于控制循环

    printf("                ----- 欢迎使用家庭财务管理系统\n");

    printf("                \n");

    printf("                1. 登录\n");

    printf("                \n");

    printf("                2. 注册\n");

    printf("                \n");

    printf("                3. 密码修改\n");
```



```

{
    struct list2 p1,*temp;
    FILE *fp;
    fp=fopen("D:\\用户信息.txt","r");
    if(fp!=NULL)
    {struct userinfo p1;
    while(fread(&p1,sizeof(struct userinfo),1,fp)==1)
    {
        temp=(struct list2 *)malloc(sizeof(struct list2));
        temp->info2=p1; //p1 为读取的用户信息的数据 放入 temp->info1
        temp->next=NULL;

        if(usehead==NULL)
        {
            usehead=temp;
            userear=temp;    //尾插法 头结点存放数据
        }
        else
        {
            userear->next=temp;
            userear=temp;
        }
    }
    fclose(fp);}
}

```

3. saveusername 函数

```

#include "head.h"
extern struct list2 *usehead;
void saveusername()
{
    struct list2 *p;
    p=usehead;
    FILE *fp;
    fp=fopen("D:\\用户信息.txt","w");
    while(p!=NULL)
    {
        fwrite(&p->info2,sizeof(struct userinfo),1,fp);
        p=p->next;//用于保存用户信息
    }
    fclose(fp);
}

```

4. login 函数

```
#include "head.h"
extern struct list2 *usehead;
extern s[20];
void login()
{
    struct list2 *q;
    q=(struct list2*)malloc(sizeof(struct list2));
    int i,k=0;//k 用于控制用户输入密码的次数
    int item,num,count=0,u;
    char s2[20]; printf("\n");
    printf("                                请输入账号:");
    scanf("%s",s);
    while(k<=3)
    {i=0;printf("\n");
    printf("                                请输入密码:");
    while(1)
    {    u=getch();
        if(u==10||u==13) //' \0' 的 ASCII 码值为 10 和 13
            break;
        printf("*");
        s2[i]=u;
        i++;
    } s2[i]='\0';
    q=usehead;
    while(q!=NULL)
    { if(strcmp(s,q->info2.username)==0&&strcmp(s2,q->info2.password)==0)
        {printf("\n\n");printf("                                登录成功!\n\n");count=1;break;}
        q=q->next;
    }
    k++;
    if(count==1){homepage();break;}
    if(count==0){printf("\n"); printf("                                密码
错误!您还有%d 次机会",3-k);}
    if(k==3){printf("\n");printf("                                输入次数
已超出限制\n\n");exit(0); }
}
}
```

5. registe 函数

```
#include "head.h"
extern struct list2 *usehead;
extern struct list2 *userear;
void registe()
```

```

{
    struct list2 *p1,*q,*p; char sh[20];
    p1=(struct list2*)malloc(sizeof(struct list2));
    int e=0,flag=1,u,pp,ff=1;
    while(flag)
    {
        flag=0;
        printf("\n                                请输入账号");
        fflush(stdin);
        scanf("%s",p1->info2.username);

        q=usehead;
        while(q!=NULL)
        { if(strcmp(p1->info2.username,q->info2.username)==0)
            {printf("\n                                已有该账号,请重新录入!\n");
              flag=1;
              break;}
          q=q->next;//保证账号不重复 (文件名不重复 造成数据混乱 )
        }

    }

    e=0;
    if(flag==0)
    {
        printf("\n                                请输入密码:");
        while((sh[e]=getch())!='\r')
        {
            printf("*");
            e++;
        }sh[e]='\0';
        e=0;
        printf("\n");
        printf("\n                                请再次输入密码:");
        while((p1->info2.password[e]=getch())!='\r')
        {
            printf("*");
            e++;
        }p1->info2.password[e]='\0';
        printf("%s",p1->info2.password);
        if(strcmp(sh,p1->info2.password)!=0)
        {printf("\n                                两次密码输入不相同,请重新
注册! ");registe();}

        printf("\n

```

```

-----\n");
printf("\n");
printf("请选择密保问题:\n");
printf("\n");
printf("1. 您父亲的姓名\n");
printf("\n");
printf("2. 您母亲的姓名\n");
printf("\n");
printf("3. 你印象最深的事\n");
printf("\n");
printf("4. 你的出身地\n");
printf("\n");
printf("5. 自定义密保密码\n");
printf("\n");
-----\n");

while(ff)
{printf("请选择:");
fflush(stdin);scanf("%d",&pp);
switch(pp)
{case 1:ff=0;printf("请输入您父亲的姓名:");
scanf("%s",p1->info2.security);break;
case 2:ff=0;printf("请输入您母亲的姓名:");
scanf("%s",p1->info2.security);break;
case 3:ff=0;printf("请输入你印象最深的事:");
scanf("%s",p1->info2.security);break;
case 4:ff=0;printf("请输入你的出身地:");
```



```

fflush(stdin);scanf("%d",&u);
switch(u)
{
case 1:ff=0;login();break;
case 2:ff=0;registe();break;
case 3:ff=0;loginmodify();break;
case 4:ff=0;exit(0);break;
default:printf("请在 1-4 之间选择\n");ff=1;
}}
}

```

6. loginmodify 函数

```

#include "head.h"
extern struct list2 *usehead;
extern s[20];
void loginmodify()
{
printf("\n\n"); FILE *fp;
fp=fopen("D:\\用户信息.txt","w");
struct list2 *q;
q=(struct list2*)malloc(sizeof(struct list2));
int i,k=0,cc;
int item,num,count=0,u,ff=1,pp,flag=0;
char s2[20],ee[20];
printf("请输入原账号:");
scanf("%s",s);
while(k<=3)
{i=0;
printf("\n请输入原密码:");
while(1)
{
u=getch();
if(u==10||u==13)
break;
printf("*");
s2[i]=u;
i++;
} s2[i]='\0';
q=usehead;
while(q!=NULL)
{ if(strcmp(s,q->info2.username)==0&&strcmp(s2,q->info2.password)==0)
{count=1;break;}
q=q->next;
}
k++;
}

```

[illegible]

```

名");
        scanf("%s", ee);
        if(strcmp(q->info2. security, ee)==0)
            {flag=1;printf("
请输入新的
密码:");
i=0;while(1)
    {    u=getch();
        if(u==10||u==13)
            break;
        printf("*");
        s2[i]=u;
        i++;
    }s2[i]='\0';
    strcpy(q->info2. password, s2);}
    saveusername();
        if(flag==0)printf("
答案错误!
\n");
        break;

        case 2:ff=0;printf("
请输入您母亲的姓名:
");
        scanf("%s", ee);
        if(strcmp(q->info2. security, ee)==0)
            {flag=1;printf("
请输入新的
密码:");
i=0;while(1)
    {    u=getch();
        if(u==10||u==13)
            break;
        printf("*");
        s2[i]=u;
        i++;
    }s2[i]='\0';
    strcpy(q->info2. password, s2);}
    saveusername();
        if(flag==0)printf("
答案错误!
\n");
        break;

        case 3:ff=0;printf("
请输入你印象最深的
事: ");
        scanf("%s", ee);
        if(strcmp(q->info2. security, ee)==0)
            {flag=1;printf("
请输入新的

```

```

密码:");
i=0;while(1)
{
    u=getch();
    if(u==10||u==13)
        break;
    printf("*");
    s2[i]=u;
    i++;
} s2[i]='\0';
strcpy(q->info2.password,s2);}
saveusername();
    if(flag==0)printf("
\n");
        break;

```

答案错误!

```

        case 4:ff=0;printf("
");
            scanf("%s",ee);
            if(strcmp(q->info2.security,ee)==0)
                {flag=1;printf("

```

请输入你的出身地:

请输入新的

```

密码:");
i=0;while(1)
{
    u=getch();
    if(u==10||u==13)
        break;
    printf("*");
    s2[i]=u;
    i++;
} s2[i]='\0';
strcpy(q->info2.password,s2);}
saveusername();
    if(flag==0)printf("
\n");
        break;

```

答案错误!

```

        case 5:ff=0;printf("
保答案: ");
            scanf("%s",ee);
            if(strcmp(q->info2.security,ee)==0)
                {flag=1;printf("

```

请输入您自定义的密

请输入新的

```

密码:");
i=0;while(1)
{
    u=getch();

```

```

        if(u==10||u==13)
            break;
        printf("*");
        s2[i]=u;
        i++;
    }s2[i]='\0';
    strcpy(q->info2.password,s2);}
    saveusername();
        if(flag==0)printf("
");
            break;

```

答案错误！

```

        default:printf("
");ff=1;break;
    }}

```

请在 1-5 之间选择！

```

    }
    if(flag==1)printf("

```

密码修改成功！\n");

```

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

```

```

        printf("
\n");

```

1. 登录

```

        printf("
\n");

```

```

        printf("
\n");

```

2. 注册

```

        printf("
\n");

```

```

        printf("
\n");

```

3. 密码修改

```

        printf("
\n");

```

```

        printf("
\n");

```

4. 退出

```

        printf("
\n");

```

```

        printf("
\n");

```

```

        printf("

```

```

-----\n");

```

```

    ff=1;
    while(ff)

```

```

{printf("
fflush(stdin);scanf("%d",&u);
switch(u)
{
case 1:ff=0;login();break;
case 2:ff=0;registe();break;
case 3:ff=0;loginmodify();break;
case 4:ff=0;exit(0);break;
default:printf("
}}
}

```

请选择: ");

请在 1-4 之间选择\n");ff=1;

7. getpassword 函数

```

#include "head.h"
extern struct list2 *usehead;
void getpassword()
{ printf("\n\n");
  struct list2 *q;
  q=(struct list2*)malloc(sizeof(struct list2));q=usehead;
  int item,num,count=0,pp,flag=0,bb,u,ff=1,i;
  char s2[20],ee[100];
  printf("
scanf("%s",s2);
  printf("
-----\n");
  printf("
\n");
  printf("
\n");
  printf("
\n");
  printf("
\n");
  printf("
\n");
  printf("
\n");
  printf("
\n");
  printf("
\n");
  printf("
\n");
}

```

请输入原账号:");

请选择待校验的密保问题:

1. 您父亲的姓名
2. 您母亲的姓名
3. 你印象最深的事

```
printf("
\n");
printf("
\n");
printf("
\n");
printf("
\n");
printf("
\n");
-----\n");
while(ff)
{
printf("                请选择:");
fflush(stdin);
scanf("%d",&pp);
switch(pp)
{case 1:ff=0;printf("请输入您父亲的姓名:");
scanf("%s",ee);
while(q!=NULL)
{ if(strcmp(s2,q->info2.username)==0&&strcmp(q->info2.security,ee)==0)
{printf("您的原密码为:%s\n",q->info2.password);
count=1;break;}
q=q->next;
}
break;

case 2:ff=0;printf("请输入您母亲的姓名:");
scanf("%s",ee);
while(q!=NULL)
{ if(strcmp(s2,q->info2.username)==0&&strcmp(q->info2.security,ee)==0)
{printf("您的原密码为:%s\n",q->info2.password);
count=1;break;}
q=q->next;
}
break;

case 3:ff=0;printf("请输入你印象最深的事:");
```



```

scanf("%s", ee);
while(q!=NULL)
{ if(strcmp(s2, q->info2. username)==0&&strcmp(q->info2. security, ee)==0)
{printf("
您的原密码
为:%s\n", q->info2. password);
count=1;break;}
q=q->next;
}
break;

```

```

case 4:ff=0;printf("
请输入你的出身地:
");
scanf("%s", ee);
while(q!=NULL)
{ if(strcmp(s2, q->info2. username)==0&&strcmp(q->info2. security, ee)==0)
{printf("
您的原密码
为:%s\n", q->info2. password);
count=1;break;}
q=q->next;
}
break;

```

```

case 5:ff=0;printf("
请输入您自定义的密保
答案:");
scanf("%s", ee);
while(q!=NULL)
{ if(strcmp(s2, q->info2. username)==0&&strcmp(q->info2. security, ee)==0)
{printf("
您的原密码
为:%s\n", q->info2. password);
count=1;break;}
q=q->next;
}
break;

```

```

default:printf("
请在 1-5 之间选择!
\n");ff=1;
}}

```

```

if(count==0){printf("
原登录信息不正
确!\n");exit(0);}
printf("

```

```
printf("\n");  
printf("1. 登录\n");  
printf("2. 注册\n");  
printf("3. 密码修改\n");  
printf("4. 退出\n");  
printf("\n");  
ff=1;  
while(ff)  
{printf("请选择：");  
fflush(stdin);scanf("%d",&u);  
switch(u)  
{  
case 1:ff=0;login();break;  
case 2:ff=0;registe();break;  
case 3:ff=0;loginmodify();break;  
case 4:ff=0;exit(0);break;  
default:printf("请在 1-4 之间选择\n");ff=1;  
}}  
}
```



```

{
    case 1:add(1);save(1);break;
    case 2:search(1);break;
    case 3:update(1);save(1);break;
    case 4:show(1);break;
    case 5:delet(1);save(1);break;
    case 6:all(1);break;
    case 7:sort(1);save(1);show(1);break;
    case 8:export(1);break;
    case 9:recycle(1);break;
    default:break;
}
if(choice>0&&choice<6)
{printf("\n");
printf("                                若继续进行收入管理 'y'/'Y'");
fflush(stdin);
scanf("%c",&ag);
}

}while(ag=='y' || ag=='Y');
}

if(i== -1)
{
do{ printf("\n\n");
printf("                                支 出 管 理\n\n");
printf("                                1. 添加                2. 查询                3. 修改\n\n");
printf("                                4. 显示                5. 删除                6. 统计\n\n");
printf("                                7. 排序                8. 导出                9. 回收站\n\n");
printf("                                请选择: ");
scanf("%d",&choice);
read(-1);
switch(choice)
{
    case 1:add(-1);save(-1);break;
    case 2:search(-1);break;
    case 3:update(-1);save(1);break;
    case 4:show(-1);break;
    case 5:delet(-1);save(-1);break;
    case 6:all(-1);break;

```

```

        case 7:sort(-1);save(-1);break;
        case 8:export(-1);break;
        case 9:recycle(-1);break;
        default:break;
    }
    if(choice>0&&choice<6)
    {
        printf("\n");
        printf("                                若继续进行支出管理 'y'/'Y' ");
        fflush(stdin);
        scanf("%c",&ag);
    }

}while(ag=='y' || ag=='Y');
}

if(i==2)
{
do{
    printf("                                统 计\n\n");
    printf("                                1. 按时间统计收入                2. 按时间统计支出\n\n");
    printf("                                3. 按成员统计收入                4. 按成员统计支出\n\n");
    printf("                                5. 按备注统计收入                6. 按备注统计支出\n\n");
    printf("                                7. 统计总收入                8. 统计总支出\n\n");
    printf("                                9. 统计当前结余                \n\n");
    printf("                                请选择:");
    scanf("%d",&choice);printf("\n\n");
    switch(choice)
    {
        case 1:read(1);all(2);break;
        case 2:read(-1);all(-2);break;
        case 3:read(1);all(3);break;
        case 4:read(-1);all(-3);break;
        case 5:read(1);all(4);break;
        case 6:read(-1);all(-4);break;
        case 7:read(1);all(1);break;
        case 8:read(-1);all(-1);break;
        case 9:read(1);money=all(1);
                read(-1);outmoney=all(-1);
    }
}
}

```

```

        printf("\n\n
", money-outmoney);
        //调用两次 计算出结余
        default:break;
    }
    if(choice>0&&choice<6)
    {
        printf("\n");
        printf("
        若继续进行统计 'y'/'Y'");
        fflush(stdin);
        scanf("%c",&ag);
    }

}while(ag=='y' || ag=='Y');
}
}

```

10. read 函数

```

#include "head.h"
extern struct list1 *head;
extern struct list1 *rear;
extern struct list1 *outhead;
extern struct list1 *outrear;
extern s[20];
extern h;
void read(int i)
{
    struct family *p1;
    struct list1 *temp;
    char file[20];
    if(i==1){if(head!=NULL)head=NULL;}
    if(i==-1){if(outhead!=NULL)outhead=NULL;}
    //如果当前头结点有指向 先初始化为 NULL 重新读取 防止多次调用 read 使节点重复

    FILE *fp;
    if(i==1)sprintf(file,"D:\\%s.收入.TXT",s);//s 为用户名 实现账户与信息一对一
    if(i==-1)sprintf(file,"D:\\%s.支出.TXT",s);
    fp=fopen(file,"r");

    while(p1=(struct family *)malloc(sizeof(struct
family )),fread(p1,sizeof(struct family ),1,fp)==1)
    {
        temp=(struct list1 *)malloc(sizeof(struct list1));
        temp->info1=*p1; //p1 为读取的数据 放入 temp->info1
    }
}

```

```

temp->next=NULL;

if(i==1)
{
if(head==NULL)
{
head=(struct list1*)malloc(sizeof(struct list1));
head->next=temp;
rear=temp;
}
else
{
rear->next=temp;
rear=temp;
}
}

if(i==1)
{
if(outhead==NULL)
{
outhead=(struct list1*)malloc(sizeof(struct list1));
outhead->next=temp;
outrear=temp;
}
else
{
outrear->next=temp;
outrear=temp;
}
}

h++; //统计读取信息的条数
}

fclose(fp);
}

```

11. save 函数

```

#include "head.h"
extern struct list1 *head;
extern struct list1 *outhead;
extern s[20];
void save(int i)
{
struct list1 *p;

```

```

char file[20];
if(head!=NULL||outhead!=NULL)
{if(i==1)p=head->next;
if(i==1)p=outhead->next;}//该 if 语句考虑到链表为空时 head->next 可能会出错
else
p=NULL;

FILE *fp;
if(i==1)sprintf(file,"D:\\%s. 收入.TXT",s);
if(i==1)sprintf(file,"D:\\%s. 支出.TXT",s);
fp=fopen(file,"wb+");

while(p!=NULL)
{
    fwrite(&p->info1,sizeof(struct family),1,fp);// 存入数据 无需存入地址
    p=p->next;
}
fclose(fp);
}

```

12. add 函数

```

#include "head.h"
extern struct list1 *head;
extern struct list1 *rear;
extern struct list1 *outhead;
extern struct list1 *outrear;
void add(int i)
{
    char ag,ch;
    struct list1 *pl;
    do
    {
        printf("\n\n");
        pl=(struct list1*)malloc(sizeof(struct list1));

        time_t rawtime;
        struct tm * timeinfo;
        time ( &rawtime );
        timeinfo = localtime ( &rawtime );
        pl->info1.year=timeinfo->tm_year+1900;
        pl->info1.month=(timeinfo->tm_mon)+1;
        pl->info1.day=timeinfo->tm_mday;//用时间函数调用系统日期

        printf("

```

日期：


```

%d-%d-%d\n\n", p1->info1.year, p1->info1.month, p1->info1.day);
    printf("                                若修改日期 'y'/'Y'\n
");
    fflush(stdin);
    ch=getchar();
    if(ch=='y')
    {printf("                                请输入日期 格式如:2001-1-1\n
");
        scanf("%d-%d-%d",&p1->info1.year,&p1->info1.month,&p1->info1.day);}
    printf("                                请输入家庭成员姓名\n
");
    fflush(stdin);
    scanf("%s", p1->info1.name);

if(i==1)
{
    printf("                                请输入收入金额\n
");
    fflush(stdin);
    scanf("%s", p1->info1.money);
}
if(i==1)
{
    printf("                                请输入支出金额\n
");
    fflush(stdin);
    scanf("%s", p1->info1.outmoney);
}

    printf("                                请输入备注\n
");
    fflush(stdin);
    scanf("%s", p1->info1.detail);

if(i==1)
{
    if(head==NULL)
    {head=(struct list*)malloc(sizeof(struct list));
    head->next=p1;//头结点为空节点便于之后的删除操作 创建链表使用尾插法
    rear=p1;
    }
    else
    {rear->next=p1;
    rear=p1;}
rear->next=NULL;

```

```

    }

    if(i==-1)
    {
        if(outhead==NULL)
        {
            outhead=(struct list1*)malloc(sizeof(struct list1));
            outhead->next=p1;
            outrear=p1;
        }
        else
        {
            outrear->next=p1;
            outrear=p1;
        }
        outrear->next=NULL;
    }

    printf("                                若继续添加 'y'/'Y' ");
    fflush(stdin);
    scanf("%c",&ag);
}while(ag=='y' || ag=='Y');
}

```

13. search 函数

```
#include "head.h"
extern struct listl *head;
extern struct listl *outhead;
void search(int i)
{
    printf("\n\n");
    char sl[20], name[2], money[20], detail[20];
    struct listl *q=NULL;
    char ag;
    int year, month, day, count=0, item, j, k, g, a=0;

    if(i==1&&head==NULL)      {printf("          \t 无任何记录! \n");a=1;}
    if(i==-1&&outhead==NULL){printf("          \t 无任何记录! \n");a=1;}

    if(a==0)
    {
        do{printf("
-----\n");

        printf("
\n");
        printf("
\n");
        printf("
\n");
```

```

\n");
    printf("
2. 按姓名查询
\n");
    printf("
3. 按金额查询
\n");
    printf("
4. 按备注查询
\n");
    printf("
5. 模糊查找
\n");
    printf("
-----\n");
    printf("                    请选择:");
    fflush(stdin);
    scanf("%d",&item);
    g=1;//g 为编号
    if(i==1)q=head->next;
    if(i==-1)q=outhead->next;
    count=0;
    switch(item)
    {
        case 1:
            printf("                    请输入查询的年月日（如：2005-02-21）:");
            fflush(stdin);
            scanf("%d-%d-%d",&year,&month,&day); printf("\n");
            printf("|N0. |t|年-月-日|t|类型|t|姓名|t|金额|t|备注|\n");
            while(q!=NULL)
            {
                if(q->info1.year==year&&q->info1.month==month&&q->info1.day==day)
                {printf("|%d|t|",g);
                    printf("%d-%d-%d|t",q->info1.year,q->info1.month,q->info1.day);
                    if(i==1)
                    {
                        printf("|收入|t|");
                        printf("%s|t|",q->info1.name);
                        printf("%s|t|",q->info1.money);
                    }
                }
                if(i==-1)

```



```

case 3:
    printf("                请输入查询的金额:");
    fflush(stdin);
    scanf("%s", money); printf("\n");
    printf("|NO. \t|年-月-日\t|类型\t|姓名\t|金额\t|备注|\n");
    while(q!=NULL)
    {
        if(strcmp(q->info1.money, money)==0)
        {
            printf("|%d\t|", g);
            printf("%d-%d-%d\t", q->info1.year, q->info1.month, q->info1.day);
            if(i==1)
            {
                printf("|收入\t|");
                printf("%s\t", q->info1.name);
                printf("%s\t", q->info1.money);
            }
            if(i==-1)
            {
                printf("|支出\t|");
                printf("%s\t", q->info1.name);
                printf("%s\t", q->info1.outmoney);
            }
            printf("%s|\n", q->info1.detail);
            g++;
        }
        q=q->next;
        count=1;
    }
    if(count==0)
    printf("                抱歉，没有该金额记录！\n"); break;

```

```

case 4:
    printf("                请输入查询的备注:");
    fflush(stdin);
    scanf("%s", detail); printf("\n");
    printf("|NO. \t|年-月-日\t|类型\t|姓名\t|金额\t|备注|\n");
    while(q!=NULL)
    {
        if(strcmp(q->info1.detail, detail)==0)
        {
            printf("|%d\t|", g);
            printf("%d-%d-%d\t", q->info1.year, q->info1.month, q->info1.day);

```

```

        if(i==1)
        {
            printf("|收入\t|");
            printf("%s\t",q->info1.name);
            printf("%s\t",q->info1.money);
        }
        if(i==-1)
        {
            printf("|支出\t|");
            printf("%s\t",q->info1.name);
            printf("%s\t",q->info1.outmoney);
        }
        printf("%s|\n",q->info1.detail);
        g++;
    }
    q=q->next;
    count=1;
}
if(count==0)
printf("                抱歉，没有该备注记录！\n");break;

case 5:
printf("                请输入查询关键字：");
scanf("%s",s1); printf("\n");
while(q)
{
    for(k=0, j=0;q->info1.name[k]!='\0' && s1[j]!='\0';)
    {

if(q->info1.name[k]==s1[j] || q->info1.detail[k]==s1[j] || q->info1.money[k]==s1[j]
)
        {
            k++;
            j++;
        }
        else
        {
            k++;
            j=0;
        }
        }//从第一个元素开始比较 比较到'\0' 若中间有一个元素不同 j 归零 重新比较
    }

if(q->info1.year==atoi(s1) || q->info1.month==atoi(s1) || q->info1.day==atoi(s1) || s1[j]=='\0')

```

```

{
    count=1;
    printf("|NO. \t|年-月-日\t|类型\t|姓名\t|金额\t|备注|\n");
    printf("|%d\t|", g);
    printf("%d-%d-%d\t", q->info1.year, q->info1.month, q->info1.day);
    if(i==1)
    {
        printf("|收入\t|");
        printf("%s\t", q->info1.name);
        printf("%s\t", q->info1.money);
    }
    if(i==-1)
    {
        printf("|支出\t|");
        printf("%s\t", q->info1.name);
        printf("%s\t", q->info1.outmoney);
    }
    printf("%s|\n", q->info1.detail);
    g++;
}
q=q->next;
} //用 count 的值来判断是否进行了上一步操作 也就是是否有关键字被找出
if(count==0)
printf("                                该关键字不存在! \n");break;

default:printf("                                请在 1-5 之间选择! \n\n");
}
printf("                                若继续查询' y' /' Y' ");
fflush(stdin);
scanf("%c", &ag);
}while(ag==' y' || ag==' Y' );
}}

```

14. Update 函数

```

#include "head.h"
extern struct list1 *head;
extern struct list1 *outhead;
void update(int i)
{
    printf("\n\n");
    struct list1 *q=NULL;
    int j, m, a=0, year, month, day, count, item, itemm, x=0;
    char ag, name[2], money[20], detail[20];

```

```

if(i==1&&head==NULL)    {printf("        \t 无任何记录! \n");a=1;}
if(i== -1&&outhead==NULL) {printf("        \t 无任何记录! \n");a=1;}

    if(a==0)
{
do{
    printf("
-----\n");

    printf("
\n");
    printf("                                1. 按日期修改
\n");
    printf("
\n");
    printf("                                2. 按姓名修改
\n");
    printf("
\n");
    printf("                                3. 按金额修改
\n");
    printf("
\n");
    printf("                                4. 按备注修改
\n");
    printf("
-----\n");

    printf("                                请选择:");
    fflush(stdin);
    scanf("%d",&item);
    j=1;
    if(i==1)q=head->next;
    if(i== -1)q=outhead->next;
    count=0;
    switch(item)
{
    case 1:
        printf("\n\n");
        printf("                                请输入要修改信息的日期(例如: 2005-02-21):");
        fflush(stdin);
        scanf("%d-%d-%d",&year,&month,&day);
        printf("NO.\t|年-月-日\t|类型\t|姓名\t|金额\t|备注\n");
        while(q!=NULL)

```



```

    { if(q->info1.year==year&&q->info1.month==month&&q->info1.day==day)
      {
        q->info1.arrange=j;//将 j 的值赋给 q->info1.arrange 方便之后找到要修改的
        那个节点
        printf("\n");
        printf("|%d\t|",q->info1.arrange);
        printf("%d-%d-%d\t",q->info1.year,q->info1.month,q->info1.day);
        if(i==1)
        {
            printf("|收入\t|");
            printf("%s\t",q->info1.name);
            printf("%s\t",q->info1.money);
        }
        if(i==1)
        {
            printf("|支出\t|");
            printf("%s\t",q->info1.name);
            printf("%s\t",q->info1.outmoney);
        }
        printf("%s|\n",q->info1.detail);
        j++;
        count=1;}
        q=q->next;
    }
    if(count==0)
        printf("                                抱歉，没有该日的财务记录！\n");
    if(count==1)
        printf("                                请输入要修改的编号：");
        scanf("%d",&m);
        printf("\n\n");
    if(i==1)q=head->next;
    if(i==1)q=outhead->next;
    while(q!=NULL)
    {
        if(m==q->info1.arrange)
        {x=1;
        printf("
\n");
        printf("                                1. 修改日期
\n");
        printf("
\n");
        printf("                                2. 修改姓名
\n");

```

```

    printf("
\n");
    printf("
\n");
    printf("
\n");
    printf("
\n");
    printf("
\n");
    fflush(stdin);
    printf("
\n");
    scanf("%d",&itemm);
    printf("\n");
    switch(itemm)
    {
        case 1:
            printf("
\n");
            fflush(stdin);
            scanf("%d-%d-%d",&q->info1.year,&q->info1.month,&q->info1.day);
            printf("\n\n");
            printf("
\n");
            printf("\n\n");
            break;

        case 2:
            printf("
\n");
            scanf("%s",q->info1.name);
            printf("\n\n");
            printf("修改成功\n");
            printf("\n\n");
            break;

        case 3:
            printf("
\n");
            if(i==1)
                scanf("%s",q->info1.money);
            if(i==1)
                scanf("%s",q->info1.outmoney);
            printf("\n\n");
            printf("
\n");
            printf("\n\n");
            break;
    }
}

```

2. 修改金额

4. 修改备注

请选择: ");

请输入新的日期 (如: 2005-02-21) :");

修改成功\n");

请输入新的姓名:");

请输入新的金额:");

修改成功\n");

```

        case 4:
            printf("                                请输入新的备注:");
            scanf("%s", q->info1.detail);
            printf("\n\n");
            printf("修改成功\n");
            printf("\n\n");
            break;

        default:printf("                                请在 1-3 之间
选择! \n");
    }

    if(x==0)printf("                                该编号不存在
\n");
    }

    q=q->next;
}
}break;

case 2:
    printf("                                请输入要修改信息的姓名:");
    fflush(stdin);
    scanf("%s", name);printf("\n\n");
    printf(" |NO. |t |年-月-日|t |类型|t |姓名|t |金额|t |备注|\n");
    while(q!=NULL)
    {
        if(strcmp(q->info1.name, name)==0)
        {
            q->info1.arrange=j;
            printf("\n");
            printf("|%d|t |", q->info1.arrange);
            printf("%d-%d-%d|t |", q->info1.year, q->info1.month, q->info1.day);
            if(i==1){printf("|收入|t |");
            printf("%s|t |", q->info1.name);
            printf("%s|t |", q->info1.money);}
            if(i==-1){printf("|支出|t |");
            printf("%s|t |", q->info1.name);
            printf("%s|t |", q->info1.outmoney);}
            printf("%s|\n", q->info1.detail);
            j++;count=1;}
        q=q->next;
    }

    if(count==0)
    printf("                                抱歉,没有该成员的财务记录!\n");

```

```

    if(count==1)
    {printf("
    scanf("%d",&m);
        if(i==1)q=head->next;
        if(i==-1)q=outhead->next;
        while(q!=NULL)
        { if(m==q->info1.arrange)
            {
                x=1;
                printf("
\n");
                printf("
\n");
                printf("
\n");
                printf("
\n");
                printf("
\n");
                printf("
\n");
                printf("
\n");
                printf("
\n");
                printf("
\n");
                printf("
\n");
                fflush(stdin);
                printf("
scanf("%d",&itemm);
printf("\n");
switch(itemm)
{
    case 1:
        printf("
        fflush(stdin);
        scanf("%d-%d-%d",&q->info1.year,&q->info1.month,&q->info1.day);
        printf("\n\n");
        printf("
        printf("\n\n");
        break;

    case 2:
        printf("
        scanf("%s",q->info1.name);

```

请输入要修改的编号：");

1. 修改日期

2. 修改姓名

2. 修改金额

4. 修改备注

请选择：");

请输入新的日期（如：2005-02-21）：");

修改成功\n");

请输入新的姓名：");

```

        printf("\n\n");
        printf("修改成功\n");
        printf("\n\n");
        break;

        case 3:
            printf("                                请输入新的金额:");
            if(i==1)
                scanf("%s", q->info1.money);
            if(i==-1)
                scanf("%s", q->info1.outmoney);
            printf("\n\n");
            printf("                                修改成功\n");
            printf("\n\n");
            break;

        case 4:
            printf("                                请输入新的备注:");
            scanf("%s", q->info1.detail);
            printf("\n\n");
            printf("修改成功\n");
            printf("\n\n");
            break;

        default:printf("                                请在 1-3 之间
选择! \n");
    }

    if(x==0)printf("                                该编号不存在
\n");
    }

    q=q->next;
}
}break;

```

```

case 3:
    printf("                                请输入姓名:");
    fflush(stdin);
    scanf("%s", money); printf("\n\n");
    printf("|NO. \t|年-月-日\t|类型\t|姓名\t|金额\t|备注|\n");
    while(q!=NULL)
    {
        if(strcmp(q->info1.money, money)==0)
        {
            q->info1.arrange=j;

```



```

        }
        if (x==0) printf("
该编号不存在\n");
    }
    q=q->next;
}
}break;

case 4:
printf("
请输入备注:");
fflush(stdin);
scanf("%s", detail); printf("\n\n");
printf("NO. \t|年-月-日\t|类型\t|姓名\t|金额\t|备注|\n");
while (q!=NULL)
{
    if (strcmp(q->info1.detail, detail)==0)
    {
        q->info1.arrange=j;
        printf("\n");
        printf("%d\t|", q->info1.arrange);
        printf("%d-%d-%d\t", q->info1.year, q->info1.month, q->info1.day);
        if (i==1) {printf("|收入\t|");
        printf("%s\t|", q->info1.name);
        printf("%s\t|", q->info1.money);}
        if (i==-1) {printf("|支出\t|");
        printf("%s\t|", q->info1.name);
        printf("%s\t|", q->info1.outmoney);}
        printf("%s|\n", q->info1.detail);
        j++;count=1;}
        q=q->next;
    }
    if (count==0)
        printf("
抱歉,没有该成员的财务记录!\n");
    if (count==1)
        {printf("
请输入要修改的编号: ");
        scanf("%d", &m);
        printf("\n\n");
        if (i==1) q=head->next;
        if (i==-1) q=outhead->next;
        while (q!=NULL)
        {
            if (m==q->info1.arrange)
            {
                x=1;

```



```

        printf("
\n");
        printf("
\n");
        printf("
\n");
        printf("
\n");
        printf("
\n");
        printf("
\n");
        printf("
\n");
        printf("
\n");
        printf("
\n");
        printf("
\n");
        printf("
\n");
        printf("
\n");
        printf("
\n");
        fflush(stdin);
        printf("
\n");
        scanf("%d",&itemm);
        printf("\n");
        switch(itemm)
        {
            case 1:
                printf("
\n");
                fflush(stdin);
                scanf("%d-%d-%d",&q->info1.year,&q->info1.month,&q->info1.day);
                printf("\n\n");
                printf("
\n");
                printf("\n\n");
                break;

            case 2:
                printf("
\n");
                scanf("%s",q->info1.name);
                printf("\n\n");
                printf("修改成功\n");
                printf("\n\n");
                break;

            case 3:
                printf("
\n");
                if(i==1)

```

1. 修改日期

2. 修改姓名

2. 修改金额

4. 修改备注

请选择:");

请输入新的日期（如：2005-02-21）:");

修改成功\n");

请输入新的姓名:");

请输入新的金额:");

```

scanf("%s", q->info1.money);
if(i==-1)
scanf("%s", q->info1.outmoney);
printf("\n\n");
printf("                                修改成功\n");
printf("\n\n");
break;

case 4:
printf("                                请输入新的备注:");
scanf("%s", q->info1.detail);
printf("\n\n");
printf("修改成功\n");
printf("\n\n");
break;

default:printf("                                请在 1-3 之间
选择! \n");
}

if(x==0)printf("                                该编号不存在
\n");
}

q=q->next;
}
}break;

default:printf("                                请在 1-4 之间选择! \n");

}

printf("                                若继续修改 'y'/'Y'");
fflush(stdin);
scanf("%c",&ag);
}while(ag=='y' || ag=='Y');
}}
```

15. show 函数

```

#include "head.h"
extern struct list1 *head;
extern struct list1 *outhead;
void show(int i)
{
printf("\n\n");
int a=0, j=1;struct list1 *p1;
if(i==1&&head==NULL) {printf("
无任何
```

```

记录! \n");a=1;}
    if(i==1&&outhead==NULL) {printf("
记录! \n");a=1;}

    if(a==0)
    {
        if(i==1)p1=head->next;//由于头结点为空节点 所以指向头结点的下一节点
        if(i==1)p1=outhead->next;
        printf("NO. \t|年-月-日\t|类型\t|姓名\t|金额\t|备注|\n");
        while(p1!=NULL)
        {
            printf("|%d\t|", j);
            printf("%d-%d-%d\t", p1->info1.year, p1->info1.month, p1->info1.day);
            if(i==1) {printf("|收入\t|");
                printf("%s\t|", p1->info1.name);
                printf("%s\t|", p1->info1.money);}
            if(i==1) {printf("|支出\t|");
                printf("%s\t|", p1->info1.name);
                printf("%s\t|", p1->info1.outmoney);}
            printf("%s|\n", p1->info1.detail);
            j++;
            p1=p1->next;
        }
    }
}

```

16. delet 函数

```

#include "head.h"
extern struct list1 *head;
extern struct list1 *outhead;
extern s[20];
extern h;
void delet(int i)
{
    printf("\n\n");
    struct list1 *q,*p;
    int a=0, year, month, day, j, m[20], count, item, itemm, x=0, num, ii;
    char ag, name[2], money[20], detail[20], file[20];

    if(i==1&&head==NULL) {printf(" \t 无任何记录! \n");a=1;}
    if(i==1&&outhead==NULL) {printf(" \t 无任何记录! \n");a=1;}

    if(a==0)
    {

```

```
do{
printf("-----\n");

printf("\n");
printf("1. 按日期删除\n");
printf("\n");
printf("2. 按姓名删除\n");
printf("\n");
printf("3. 按金额删除\n");
printf("\n");
printf("4. 按备注删除\n");
printf("\n");
printf("5. 一键删除所有记录\n");
printf("-----\n");

printf("请选择:");
fflush(stdin);
scanf("%d",&item);
printf("\n");
if(i==1)q=head->next;
if(i==-1)q=outhead->next;
count=0;j=1;
switch(item)
{
case 1:
printf("请输入要删除信息的日期（格式：2005-02-21）:");
fflush(stdin);
scanf("%d-%d-%d",&year,&month,&day);
printf("NO.\t|年-月-日\t|类型\t|姓名\t|金额\t|备注|\n\n");
while(q!=NULL)
{ if(q->info1.year==year&&q->info1.month==month&&q->info1.day==day)
{
```

q->info1.arrange=j;//将 j 的值赋给 q->info1.arrange 方便之后找到要删除的那个节点

```
printf("\n");
printf("          |%d\t|", q->info1.arrange);
printf("%d-%d-%d\t", q->info1.year, q->info1.month, q->info1.day);
if(i==1){printf("|收入\t|");
printf("%s\t", q->info1.name);
printf("%s\t", q->info1.money);}
if(i==-1){printf("|支出\t|");
printf("%s\t", q->info1.name);
printf("%s\t", q->info1.outmoney);}
printf("%s|\n\n", q->info1.detail);
j++;
count=1;
}
q=q->next;
}
if(count==0)
printf("          抱歉，没有该日的财务记录！\n");

if(count==1)
{

printf("          请输入要删除的数量：");

scanf("%d", &num); if(num>j)printf("删除的数量超过可删除数量！");
printf("          请输入要删除的编号：");
for(ii=0;ii<num;ii++)
{scanf("%d", &m[ii]);};//用数组 m 记录下要删除的编号 与之后的 q->info1.arrange
去匹配 找出要删除的节点
printf("\n          ");
ii=0;
while(ii<num)//循环进行的次数为要删除数量的次数
{
if(m[ii]<j)
{if(i==1)q=head;
if(i==-1)q=outhead;
while(q!=NULL)
{ p=q;
q=q->next;
if(m[ii]==q->info1.arrange)//节点编号与 m 数组中所记录的编号值相同 则
进行删除操作
{
```

情况

```
p->next=q->next;//由于头结点为空 无需考虑要删除的节点为头结点的
```

```
FILE *fp;
    if(i==1)sprintf(file,"D:\\%s. 收入回收站.TXT",s);
    if(i==-1)sprintf(file,"D:\\%s. 支出回收站.TXT",s);
    fp=fopen(file,"a+");
    fwrite(&q->info1,sizeof(struct family),1,fp);
    fclose(fp);//将该节点数据保存到 该用户对应的回收站文件中

    break;
}
}}ii++;
}

}        printf("        删除成功\n");
        printf("        温馨提示: 如果误删信息可以去回收站找\n");
break;
```

case 2:

```
printf("        请输入要删除信息的姓名:");
fflush(stdin);
scanf("%s",name);
printf("\n");
printf("        |NO.\t|年-月-日\t|类型\t|姓名\t|金额\t|备注|\n\n");
while(q!=NULL)
{
    if(strcmp(q->info1.name,name)==0)
    {
        q->info1.arrange=j;
        printf("        |%d\t|",q->info1.arrange);
        printf("%d-%d-%d\t",q->info1.year,q->info1.month,q->info1.day);
        if(i==1)
        {printf("|收入\t|");
        printf("%s\t|",q->info1.name);
        printf("%s\t|",q->info1.money);
        }
        if(i==-1)
        {printf("|支出\t|");
        printf("%s\t|",q->info1.name);
        printf("%s\t|",q->info1.outmoney);
        }
    }
}
```

```

        printf("%s|\n\n", q->info1.detail);
        j++;
        count=1;
    }
    q=q->next;

}
if(count==0)
printf("          抱歉，没有该姓名的财务记录！\n");
    if(count==1)
{

    printf("          请输入要删除的数量：");
    scanf("%d",&num);
    printf("          请输入要删除的编号：");
    for(ii=0;ii<num;ii++)
    {scanf("%d",&m[ii]);
    printf("\n          ");}
    ii=0;
    while(ii<num)
    { if(m[ii]<j)
        {if(i==1)q=head;
        if(i==-1)q=outhead;
        while(q!=NULL)
        { p=q;
        q=q->next;
        if(m[ii]==q->info1.arrange)
        {

            p->next=q->next;
            FILE *fp;
            if(i==1)sprintf(file,"D:\\%s.收入回收站.TXT",s);
            if(i==-1)sprintf(file,"D:\\%s.支出回收站.TXT",s);
            fp=fopen(file,"a+");
            fwrite(&q->info1,sizeof(struct family),1,fp);
            fclose(fp);

            break;
        }
        }}ii++;
    }
}

```

```

    }          printf("          删除成功\n");
               printf("          温馨提示：如果误删信息可以去回收站找\n");
    break;

```

case 3:

```

    printf("          请输入要删除信息的金额:");
    fflush(stdin);
    scanf("%s", money);
    printf("\n");
    printf("          |NO.\t|年-月-日\t|类型\t|姓名\t|金额\t|备注|\n\n");
    while(q!=NULL)
    {   if(strcmp(q->info1.money, money)==0)
        {   q->info1.arrange=j;
            printf("          |%d\t|", q->info1.arrange);
            printf("%d-%d-%d\t", q->info1.year, q->info1.month, q->info1.day);
            if(i==1)
            {printf("|收入\t|");
              printf("%s\t|", q->info1.name);
              printf("%s\t|", q->info1.money);
            }
            if(i== -1)
            {printf("|支出\t|");
              printf("%s\t|", q->info1.name);
              printf("%s\t|", q->info1.outmoney);
            }
            printf("%s|\n\n", q->info1.detail);
            j++;
            count=1;
        }
        q=q->next;
    }
    if(count==0)
    printf("          抱歉，没有该金额的财务记录！\n");
    if(count==1)
    {

        printf("          请输入要删除的数量:");
        scanf("%d", &num);
        printf("          请输入要删除的编号:");
        for(ii=0; ii<num; ii++)
        {scanf("%d", &m[ii]);
          printf("\n          ");
        }
    }
}

```



```

        ii=0;
while(ii<num)
{ if(m[ii]<j)
    {if(i==1)q=head;
     if(i==1)q=outhead;
while(q!=NULL)
    { p=q;
      q=q->next;
      if(m[ii]==q->info1.arrange)
      {

          p->next=q->next;
          FILE *fp;
          if(i==1)sprintf(file,"D:\\%. 收入回收站.TXT",s);
          if(i==1)sprintf(file,"D:\\%. 支出回收站.TXT",s);
          fp=fopen(file,"a+");
          fwrite(&q->info1,sizeof(struct family),1,fp);
          fclose(fp);

          break;
      }
    }}ii++;
}

}          printf("          删除成功\n");
          printf("          温馨提示：如果误删信息可以去回收站找\n");
break;

case 4:
    printf("          请输入要删除信息的备注:");
    fflush(stdin);
    scanf("%s",detail);
    printf("\n");
    printf("          |NO.\t|年-月-日\t|类型\t|姓名\t|金额\t|备注|\n\n");
    while(q!=NULL)
    { printf("\n");
      if(strcmp(q->info1.detail,detail)==0)
      {
          q->info1.arrange=j;
          printf("          |%d\t|",q->info1.arrange);
          printf("%d-%d-%d\t",q->info1.year,q->info1.month,q->info1.day);
          if(i==1)

```

```

        {printf("|收入\t|");
        printf("%s\t|", q->infol.name);
        printf("%s\t|", q->infol.money);
        }
        if(i==1)
        {printf("|支出\t|");
        printf("%s\t|", q->infol.name);
        printf("%s\t|", q->infol.outmoney);
        }
        printf("%s|\n\n", q->infol.detail);
        j++;
        count=1;
    }
    q=q->next;

}
if(count==0)
printf("          抱歉，没有该备注的财务记录！\n");
    if(count==1)
{

    printf("          请输入要删除的数量：");
    scanf("%d",&num);
    printf("          请输入要删除的编号：");
    for(ii=0;ii<num;ii++)
    {scanf("%d",&m[ii]);
    printf("\n          ");}
    ii=0;
    while(ii<num)
    { if(m[ii]<j)
        {if(i==1)q=head;
        if(i==1)q=outhead;
        while(q!=NULL)
        { p=q;
        q=q->next;
        if(m[ii]==q->infol.arrange)
        {

            p->next=q->next;
            FILE *fp;
            if(i==1)sprintf(file,"D:\\%s.收入回收站.TXT",s);
            if(i==1)sprintf(file,"D:\\%s.支出回收站.TXT",s);
            fp=fopen(file,"a+");
            fwrite(&q->infol,sizeof(struct family),1,fp);

```

```

        fclose(fp);

        break;
    }
    }}ii++;
}

}        printf("        删除成功\n");
        printf("        温馨提示：如果误删信息可以去回收站找\n");
        break;

    case 5:
        if(i==1)q=head->next;
        if(i==1)q=outhead->next;

FILE *fp;
if(i==1)sprintf(file,"D:\\%.s. 收入回收站.TXT",s);
if(i==1)sprintf(file,"D:\\%.s. 支出回收站.TXT",s);
fp=fopen(file,"wb+");

while(q!=NULL)
{
    fwrite(&q->info1,sizeof(struct family),1,fp);
    q=q->next;
}
fclose(fp);
// 由于是一键删除 先把所有数据存入 该用户对应的回收站文件中
if(i==1)head=NULL;
if(i==1)outhead=NULL;
//直接让头结点指向 NULL 链表即为空
        break;

    default:printf("        请在 1-5 之间选择! \n\n");
}
printf("        若继续删除'y'/'Y'");
fflush(stdin);
scanf("%c",&ag);
}while(ag=='y' || ag=='Y');
}
}

```

17.all 函数

```
#include "head.h"
```

```

extern struct list1 *head;
extern struct list1 *outhead;
float all(int i)
{
    float money=0;
    struct list1 *q;
    int count=0, year, month, day, a=0;
    char name[20], detail[20];
    if(i==1&&head==NULL) {printf("          \t 无任何记录! \n");a=1;}
    if(i== -1&&outhead==NULL) {printf("          \t 无任何记录! \n");a=1;}
    if(a==0)
    {
        if(i==1)q=head->next;
        if(i== -1)q=outhead->next;
        printf("\n\n");
    }
    if(i==1 || i== -1)
    {
        while(q)
        {
            if(i==1)money+=atoi(q->info1.money);
            if(i== -1)money+=atoi(q->info1.outmoney); //atoi 函数将字符型转换为浮点型 用于金额总数的统计
            q=q->next;
        }
        if(i==1)printf("          总收入为: %.2f 元\n", money);
        if(i== -1)printf("          总支出为: %.2f 元\n", money);
    }

    if(i==2 || i== -2)
    { printf("请输入查询的年月日 (例如: 2005-02-21) :");
      fflush(stdin);
      scanf("%d-%d-%d", &year, &month, &day);
      while(q!=NULL)
      { if(q->info1.year==year&&q->info1.month==month&&q->info1.day==day)
        {if(i==2)money+=atoi(q->info1.money);
         if(i== -2)money+=atoi(q->info1.outmoney);
          count=1;}
        q=q->next;
      }
      if(count==0)
      printf("抱歉, 没有该日信息记录! \n");
      if(i==2)printf("%d-%d-%d 总收入为: %.2f 元\n", year, month, day, money);
      if(i== -2)printf("%d-%d-%d 总支出为: %.2f 元\n", year, month, day, money);
    }
}

```

```

}

    if(i==3||i==--3)
    {
        printf("请输入查询的姓名:");
        fflush(stdin);
        scanf("%s",name);
        q=head->next;
        while(q!=NULL)
        {
            if(strcmp(q->info1.name,name)==0)
            {
                if(i==3)money+=atoi(q->info1.money);
                if(i==--3)money+=atoi(q->info1.outmoney);}
            q=q->next;
            count=1;
        }
        if(count==0)
        printf("抱歉，没有该姓名记录! \n");
        if(i==3)printf("家庭成员: %s 的总收入为: %.2f 元\n",name,money);
        if(i==--3)printf("家庭成员: %s 的总支出为: %.2f 元\n",name,money);

    }

```

```

        if(i==4||i==--4)
        {
            printf("请输入查询的姓名:");
            fflush(stdin);
            scanf("%s",detail);
            q=head->next;
            while(q!=NULL)
            {
                if(strcmp(q->info1.detail,detail)==0)
                {
                    if(i==4)money+=atoi(q->info1.money);
                    if(i==--4)money+=atoi(q->info1.outmoney);}
                q=q->next;
                count=1;
            }
            if(count==0)
            printf("抱歉，没有该姓名记录! \n");
            if(i==4)printf("备注: %s 的总收入为: %.2f 元\n",detail,money);
            if(i==--4)printf("备注: %s 的总支出为: %.2f 元\n",detail,money);

        }

```

}return money;//返回金额用于统计结余 先调用 all(1)算出总收入 在调用 all(-1)算出总支出 二者相减得结余

18. sort 函数

```
#include "head.h"
extern struct listl *head;
extern struct listl *outhead;
extern h;
void sort(int i)
{   printf("\n\n");
    struct listl *q,*p,*f,*temp,*m,*pl;
    m=head;
    int a=0,b,j,item;;char ag;
    if(i==1&&head==NULL){printf("          \t 无任何记录! \n");a=1;}
    if(i==-1&&outhead==NULL){printf("          \t 无任何记录! \n");a=1;}
        if(a==0)
            {
do{printf("-----\n");

        printf("
\n");
        printf("                                1. 按日期排序
\n");
        printf("
\n");
        printf("                                2. 按姓名排序
\n");
        printf("
\n");
        printf("                                3. 按金额排序
\n");
        printf("
\n");
        printf("                                4. 按备注排序
\n");
        printf("
\n");
        printf("-----\n");

        printf("                    请选择:") ;
        fflush(stdin);
        scanf("%d",&item);b=0;
```

```

        if(i==1)q=head;
        if(i==1)q=outhead;
        j=1;
        switch(item)
        {

case 1:
        while(b<=h)
        {q=m;
            while(q->next&&q->next->next)
            {f=q;
                p=q->next;
                q=p->next;
                if(p->info1.year>q->info1.year)
                {
                    p->next=q->next;
                    q->next=p;
                    f->next=q;
                }else q=p;
            }
            //按时间排序 先判断年份 年份小的排前面 再判断月份和日期
            //如果年份排序正确 直接比较月份 以此类推
            //如果没有交换位置要将 q=p 进行下个位置的比较 否则中间会漏掉节点
        }
        b++;
    }
    if(i==1)q=head;
    if(i==1)q=outhead;
    b=0;
    while(b<=h)
    {q=m;
        while(q->next&&q->next->next)
        {f=q;
            p=q->next;
            q=p->next;
            if(p->info1.year==q->info1.year&&p->info1.month>q->info1.month)
            {
                p->next=q->next;
                q->next=p;
                f->next=q;
            }else q=p;
        }
        b++;
    }
    if(i==1)q=head;

```

```

    if(i==1)q=outhead;
    b=0;
    while(b<=h)
    {q=m;
        while(q->next&&q->next->next)
        {f=q;
            p=q->next;
            q=p->next;

            if(p->info1.year==q->info1.year&&p->info1.month==q->info1.month&&p->info1.d
ay>q->info1.day)
            {
                p->next=q->next;
                q->next=p;
                f->next=q;
            } else q=p;
        }
        b++;
    }

    if(i==1)p1=head->next;
    if(i==1)p1=outhead->next;
    printf("|NO. \t|年-月-日\t|类型\t|姓名\t|金额\t|备注|\n");
    while(p1!=NULL)
    {
        printf("|%d\t|",j);
        printf("%d-%d-%d\t",p1->info1.year,p1->info1.month,p1->info1.day);
        if(i==1)
        {printf("|收入\t|");
            printf("%s\t",p1->info1.name);
            printf("%s\t",p1->info1.money);
        }
        if(i==1)
        {printf("|支出\t|");
            printf("%s\t",p1->info1.name);
            printf("%s\t",p1->info1.outmoney);
        }
        printf("%s|\n",p1->info1.detail);
        j++;
        p1=p1->next;
    }

    break;

```



```

case 2:
    while (b<=h)
    {
        q=m;
        while (q&&q->next&&q->next->next)
        {f=q;
            p=q->next;
            q=p->next;
            if (strcmp(p->infol.name, q->infol.name)>0)
            {
                p->next=q->next;
                q->next=p;
                f->next=q;
            }
            else q=p;
            } b++;
        }
        if (i==1) p1=head->next;
        if (i==-1) p1=outhead->next;
        printf("NO. \t | 年-月-日 \t | 类型 \t | 姓名 \t | 金额 \t | 备注 | \n");
        while (p1!=NULL)
        {
            printf("| %d \t |", j);
            printf("%d-%d-%d \t", p1->infol.year, p1->infol.month, p1->infol.day);
            if (i==1) {printf("| 收入 \t |");
                printf("%s \t |", p1->infol.name);
                printf("%s \t |", p1->infol.money);}
            if (i==-1) {printf("| 支出 \t |");
                printf("%s \t |", p1->infol.name);
                printf("%s \t |", p1->infol.outmoney);}
            printf("%s | \n", p1->infol.detail);
            j++;
            p1=p1->next;
        }
        break;

    case 3:
        while (b<=h)
        {
            q=m;
            while (q&&q->next&&q->next->next)
            {f=q;
                p=q->next;

```

```

        q=p->next;
    if(i==1)
        {if(atoi(p->info1.money)>atoi(q->info1.money))//先将金额由字符型转化为数值
型再进行排序
        {
            p->next=q->next;
            q->next=p;
            f->next=q;
        }
        else q=p;}

    if(i==-1)
        {if(atoi(p->info1.outmoney)>atoi(q->info1.outmoney))
        {
            p->next=q->next;
            q->next=p;
            f->next=q;
        }
        else q=p;}

    }b++;
    }
    if(i==1)p1=head->next;
    if(i==-1)p1=outhead->next;
    printf("|NO. \t|年-月-日\t|类型\t|姓名\t|金额\t|备注|\n");
    while(p1!=NULL)
    {
        printf("|%d\t|", j);
        printf("%d-%d-%d\t", p1->info1.year, p1->info1.month, p1->info1.day);
        if(i==1){printf("|收入\t|");
            printf("%s\t|", p1->info1.name);
            printf("%s\t|", p1->info1.money);}
        if(i==-1){printf("|支出\t|");
            printf("%s\t|", p1->info1.name);
            printf("%s\t|", p1->info1.outmoney);}
        printf("%s|\n", p1->info1.detail);
        j++;
        p1=p1->next;
    }
    break;

    case 4:
    while(b<=h)
    {

```

```

    q=m;
    while(q&&q->next&&q->next->next)
    {f=q;
      p=q->next;
      q=p->next;
      if(strcmp(p->info1.detail,q->info1.detail)>0)
      {
        p->next=q->next;
        q->next=p;
        f->next=q;
      }
    }
    else q=p;
    }b++;
  }
  if(i==1)p1=head->next;
  if(i==-1)p1=outhead->next;
  printf("|NO. \t|年-月-日\t|类型\t|姓名\t|金额\t|备注|\n");
  while(p1!=NULL)
  {
    printf("|%d\t|",j);
    printf("%d-%d-%d\t",p1->info1.year,p1->info1.month,p1->info1.day);
    if(i==1){printf("|收入\t|");
      printf("%s\t|",p1->info1.name);
      printf("%s\t|",p1->info1.money);}
    if(i==-1){printf("|支出\t|");
      printf("%s\t|",p1->info1.name);
      printf("%s\t|",p1->info1.outmoney);}
    printf("%s|\n",p1->info1.detail);
    j++;
    p1=p1->next;
  }
  break;
}
printf("                                若继续排序'y'/'Y' ");
fflush(stdin);
scanf("%c",&ag);
}while(ag=='y' || ag=='Y');
}
}

```

19. export 函数

```

#include "head.h"
extern struct list1 *head;
extern struct list1 *outhead;
extern s[20];

```

```

void export(int i)
{
    char file[20];
    if(i==1)sprintf(file, "%s. 收入信息.csv", s);
    if(i== -1)sprintf(file, "%s. 支出信息.csv", s); //导出到该用户名对应的文件中
    FILE * fp=fopen(file, "w");
    fprintf(fp, "%s %s %s %s %s %s\n", "年", "月", "日", "姓名", "金额", "备注");
    struct list1 *p;
    if(i==1)p=head->next;
    if(i== -1)p=outhead->next;
    while(p!=NULL)
    {
        if(i==1)
            fprintf(fp, "%d %d %d %s %s %s\n", p->info1.year, p->info1.month, p->info1.day, p->info1.name, p->info1.money, p->info1.detail);
        if(i== -1)
            fprintf(fp, "%d %d %d %s %s %s\n", p->info1.year, p->info1.month, p->info1.day, p->info1.name, p->info1.outmoney, p->info1.detail);
        p=p->next;
    }
    fclose(fp);
    printf("\n\n          信息已成功导出\n\n");
}

```

20. recycle 函数

```

#include "head.h"
extern s[20];
void recycle(int i)
{
    struct list1 *temp, *recyclehead=NULL, *recyclerear=NULL, *p, *q;
    printf("\n\n"); char file[20];
    struct family *p1;
    FILE *fp; int j=1; char ch; int num, ii, m[30];
    if(i==1)sprintf(file, "D:\\%s. 收入回收站.TXT", s); //打开该用户对应的回收站文件进行读取
    if(i== -1)sprintf(file, "D:\\%s. 支出回收站.TXT", s);
    fp=fopen(file, "a+");
    while(p1=(struct family *)malloc(sizeof(struct family)),
        fread(p1, sizeof(struct family), 1, fp)==1)
    {
        temp=(struct list1 *)malloc(sizeof(struct list1));

```

```

        temp->infol=*p1;
        temp->next=NULL;

    if(recyclehead==NULL)
    {
        recyclehead=(struct list1*)malloc(sizeof(struct list1));
        recyclehead->next=temp;
        recyclerear=temp;
    }
    else
    {
        recyclerear->next=temp;
        recyclerear=temp;
    }

}

p=recyclehead->next;
while(p)
{
    p->infol.arrange=j;
    printf("|%d\t|", j);
    printf("%d-%d-%d\t", p->infol.year, p->infol.month, p->infol.day);
    if(i==1){printf("|收入\t|");
        printf("%s\t", p->infol.name);
        printf("%s\t", p->infol.money);}
    if(i==-1){printf("|支出\t|");
        printf("%s\t", p->infol.name);
        printf("%s\t", p->infol.outmoney);}
    printf("%s|\n", p->infol.detail);
    j++;
    p=p->next;
}

printf("                是否进行恢复信息操作 Y/y");
fflush(stdin);
scanf("%c",&ch);
if(ch=='y' || ch=='Y')
{
    printf("                请输入要恢复信息的数量：");
    scanf("%d",&num);
    printf("                请输入要恢复的编号：");
    for(ii=0;ii<num;ii++)
    {fflush(stdin);

```

```
scanf("%d",&m[ii]);
printf("\n          ");}
ii=0;
while(ii<num)
{ if(m[ii]<j)
    {
        q=recyclehead;
        while(q!=NULL)
        { p=q;
            q=q->next;

            if(m[ii]==q->info1.arrange)
            {

                p->next=q->next;
                FILE *ffp;
                if(i==1)sprintf(file,"D:\\%s.收入.TXT",s);
                if(i==-1)sprintf(file,"D:\\%s.支出.TXT",s);
                ffp=fopen(file,"a+");
                fwrite(&q->info1,sizeof(struct family),1,ffp);
                free(q);
                fclose(ffp);
                break;
            }
        }
        ii++;
    }
}
```

//将恢复的信息从回收站删除 并存入原文件中

```
q=recyclehead->next;
if(i==1)sprintf(file,"D:\\%s.收入回收站.TXT",s);
if(i==-1)sprintf(file,"D:\\%s.支出回收站.TXT",s);
fp=fopen(file,"w");
while(q)
{
    fwrite(&q->info1,sizeof(struct family),1,fp);
    q=q->next;
}
fclose(fp); }
```

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <conio.h>
```

```
#include <math.h>
#include <time.h>
struct family
{
    int year;
    int month;
    int day;
    char name[20];
    char money[20];
    char outmoney[20];
    char detail[20];
    int  arrange;
};
struct list1
{
    struct family info1;
    struct list1 *next;
};
struct userinfo
{
    char username[10];
    char password[10];
    char security[10];
};
struct list2
{
    struct userinfo info2;
    struct list2 *next;
};
void readusername();
void saveusername();
void login();
void registe();
void loginmodify();
void getpassword();
void homepage();
void menu(int i);
void read(int i);
void save(int i);
void add(int i);
void search(int i);
void update(int i);
void show(int i);
void delet(int i);
```

```
float all(int i);  
void sort(int i);  
void export(int i);  
void recycle(int i);
```