

肇庆学院



专业技能训练报告

学 院： 计算机软件学院

课程名称： C 程序设计综合实践

专业班级： 软件工程 16 软件 2 班

学生姓名： 何承翰 学号： 201624133225

指导教师： 赵玉明老师

完成时间： 2017 年 月 日

目录

第 1 部分 实训题目与要求	1
1. 问题提出.....	1
2. 功能要求.....	1
3. 任务分工.....	2
第 2 部分 设计实训题目功能	3
1. 总体设计.....	3
2. 算法设计.....	4
1. Input_Choice.....	4
2. Build_File.....	4
3. Creat_new	4
4. *Start_Card	5
5. Insert_Card.....	5
6. Use_Card.....	5
7. Charge_Money	6
8. Delete_Card	6
9. Lost_Card.....	6
10. Search_Card	7
11. Sort_Card	7
12. Print_Card	7
13. Open_File&Close_File	8
3. 数据结构.....	8
4. 程序代码设计.....	8
1. Input_Choice.....	8
2. Build_File.....	8
3. Creat_new	9
4. Start_Card	9
5. Insert_Card.....	9
6. Use_Card.....	9
7. Charge_Money	9
8. Delete_Card	9
9. Lost_Card.....	9
10. Search_Card	10
11. Sort_Card	10
12. Print_Card	10
13. Open_File&Close_File	10
5. 测试与调试.....	10
第 3 部分 实训总结	11
1. 个人总结.....	11
2. 结束语.....	11
附录 A 程序清单.....	12

附录 B 用户使用说明书.....	20
参考文献	20
《C 程序设计综合实践》评分表	21

第 1 部分 实训题目与要求

1. 问题提出

饭堂用饭卡卖饭，不仅提高卖饭速度，还通过计算机管理，提高了管理效率和管理质量。

2. 功能要求

重复显示如图 1 所示的主菜单，在主菜单中选择任意一项，均实现相应功能。

```
-----
☆  请输入选项编号（0-9）：  ☆
-----
★    1--建立饭卡文件      ★
★    2--消费              ★
★    3--充值              ★
★    4--添加新饭卡        ★
★    5--注销旧饭卡        ★
★    6--设置与解除挂失    ★
★    7--查询饭卡信息      ★
★    8--排序饭卡信息      ★
★    9--遍历饭卡信息      ★
★    0--退出系统          ★
-----
```

图 1 饭卡管理主菜单

在主菜单中选择 1，建立名为 `card.dat` 的文件，并在其中添加若干人的饭卡号、姓名、金额、挂失信息，要求饭卡号是唯一的。

在主菜单中选择 2，要求用户输入饭卡号、饭费，然后系统自动从该人的饭卡中减去饭钱，并分别显示买饭前后的金额，如果原来饭卡中的余额不足 5 元，则不能买饭，而且显示“余额不够，请续钱！”，如果挂失信息为真，则显示“本卡已冻结！”。

在主菜单中选择 3，要求用户输入饭卡号、续钱额，续钱完成后分别显示续钱前后的金额。

在主菜单中选择 4，添加新饭卡，这时要求新饭卡号和已有的饭卡号不能重号。

在主菜单中选择 5，注销旧饭卡。

在主菜单中选择 6，要求用户输入饭卡号和挂失信息，然后更新该饭卡的挂失信息。

在主菜单中选择 7、8、9，此项功能学生自由发挥，根据本组爱好增加与本题目相关的新功能。

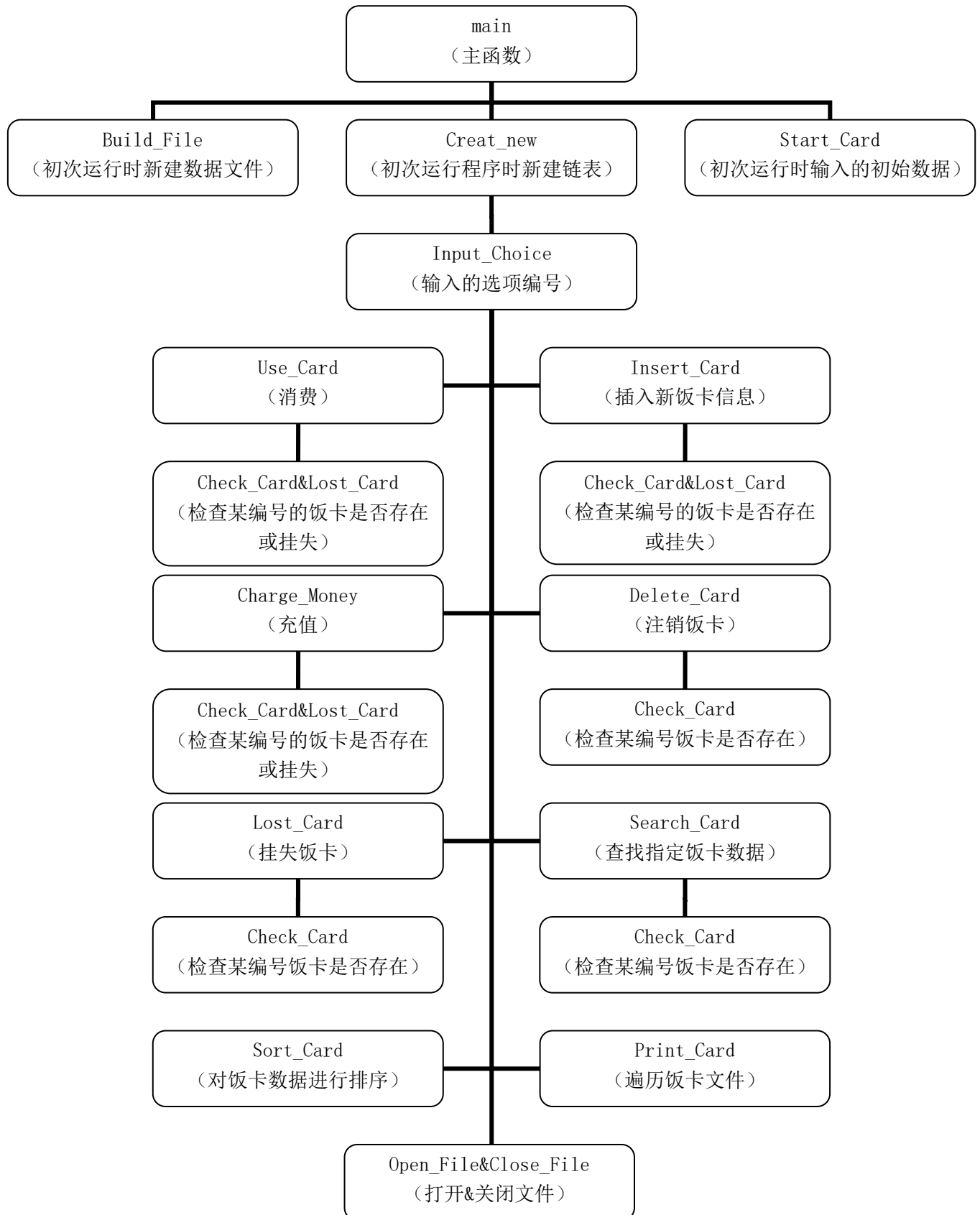
在主菜单中选择 0，显示结束信息（如“感谢使用本软件！已正常退出，按任意键结束！”），按任意键后，退出本系统。

3. 任务分工

一人全包。

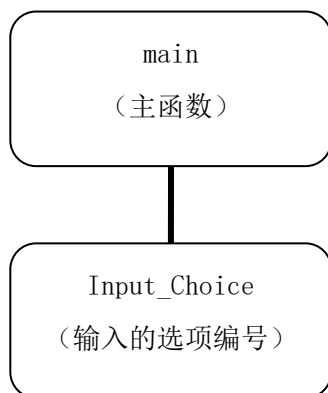
第2部分 设计实训题目功能

1. 总体设计

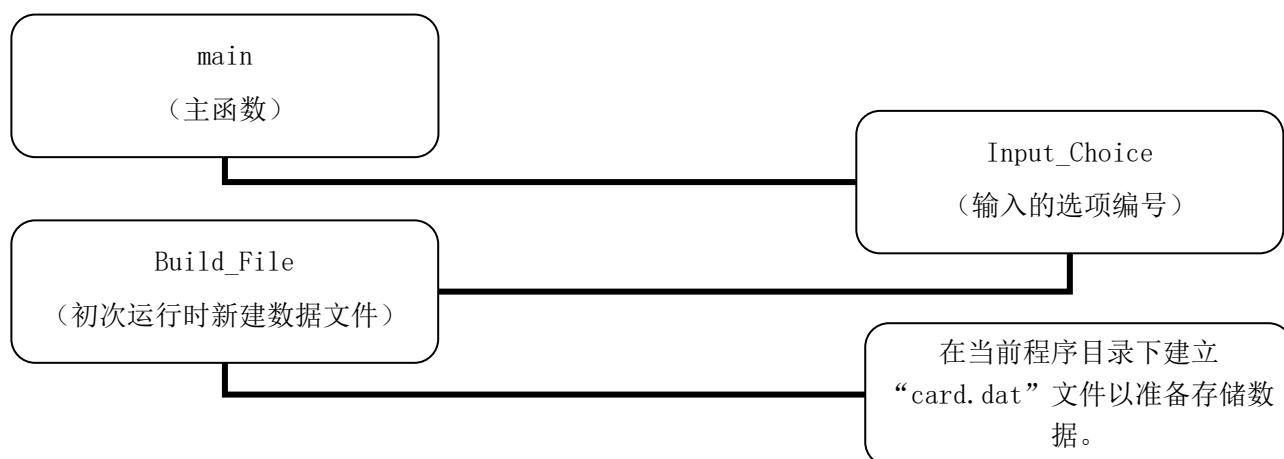


2. 算法设计

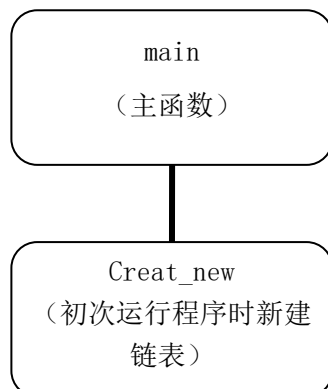
1. Input_Choice



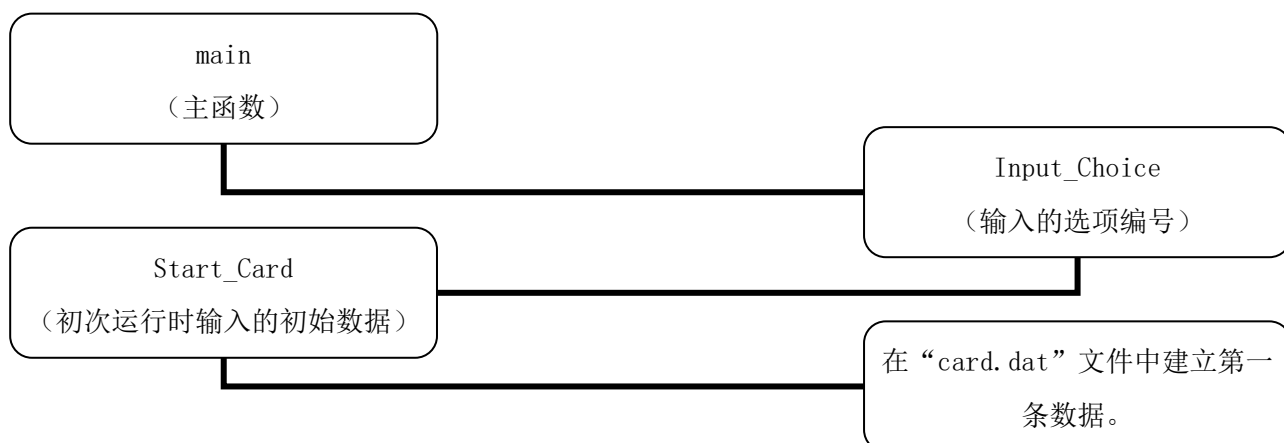
2. Build_File



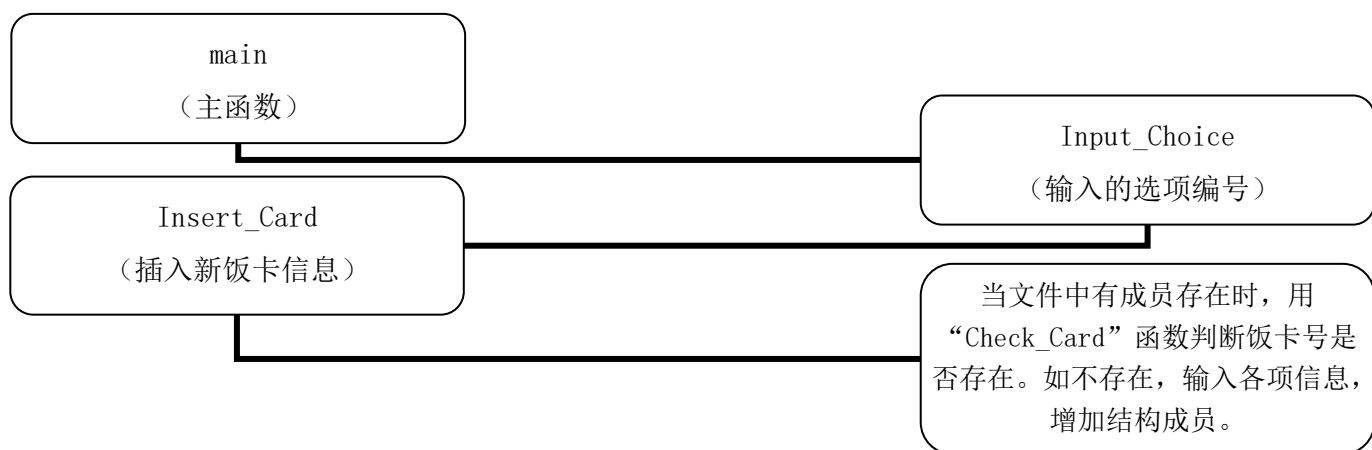
3. Creat_new



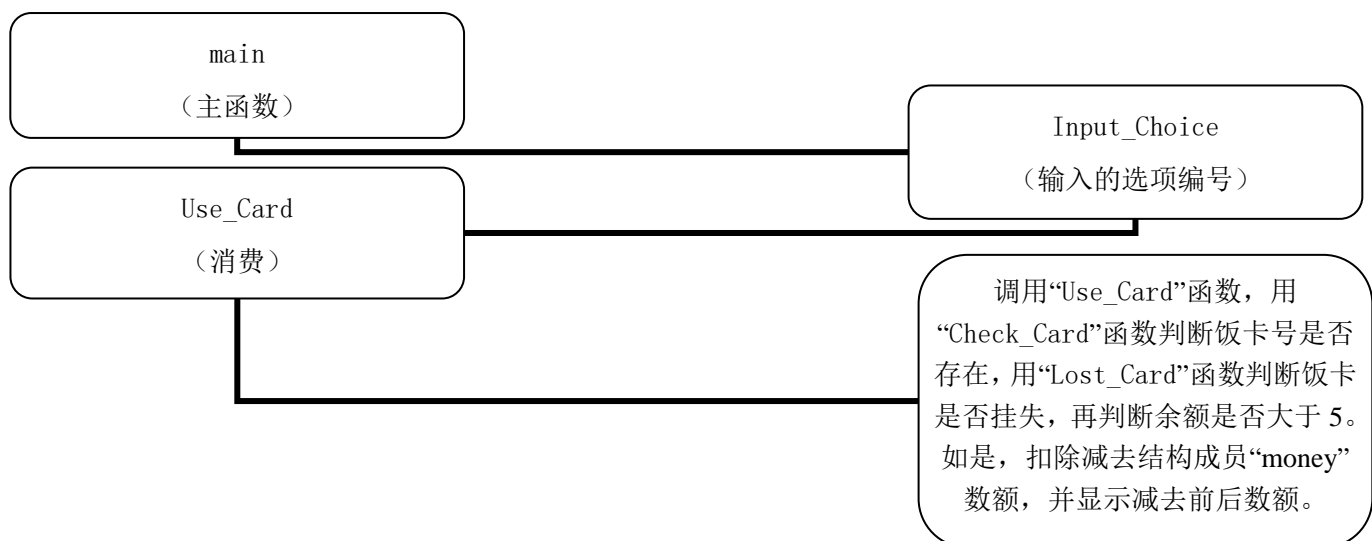
4. *Start_Card



5. Insert_Card

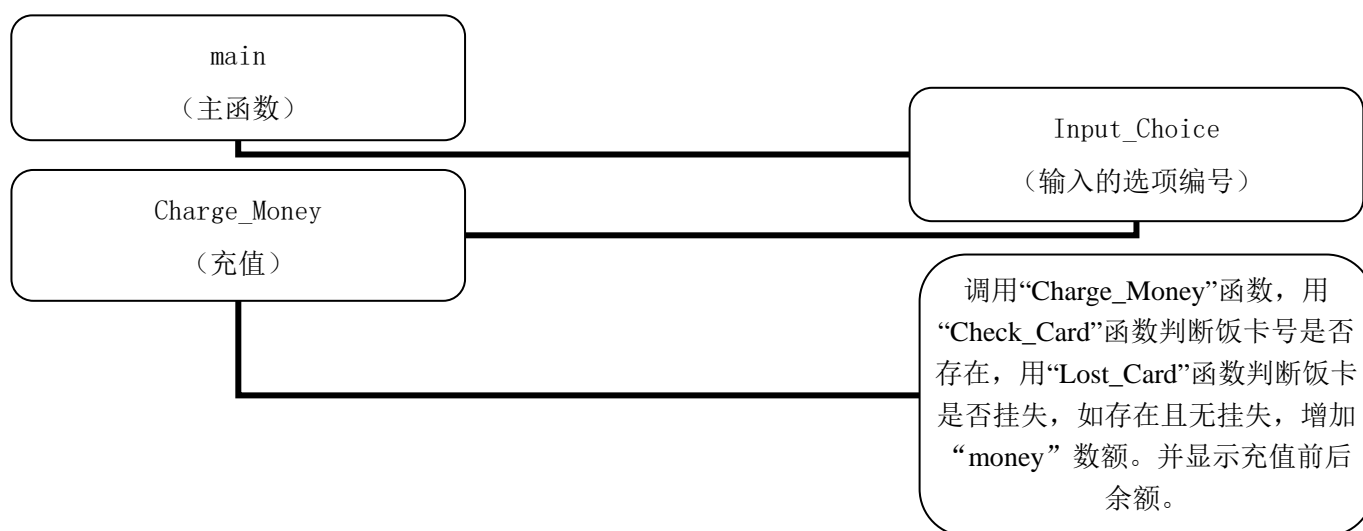


6. Use_Card

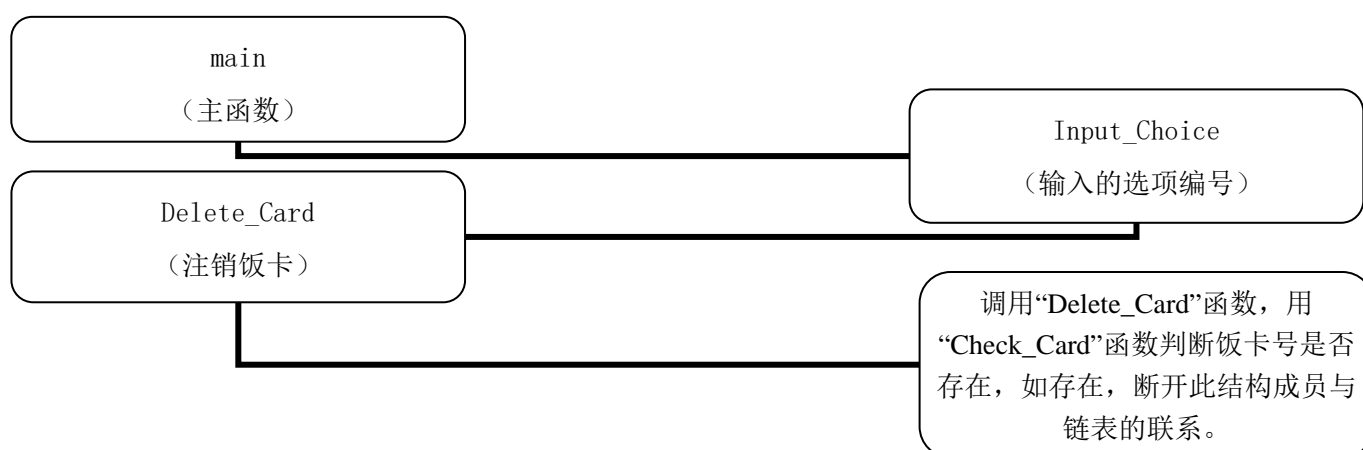


*由于初次执行程序时会出现群龙无首的错误，所以把「第一次接收数据」和「第N次接收数据」分类讨论。

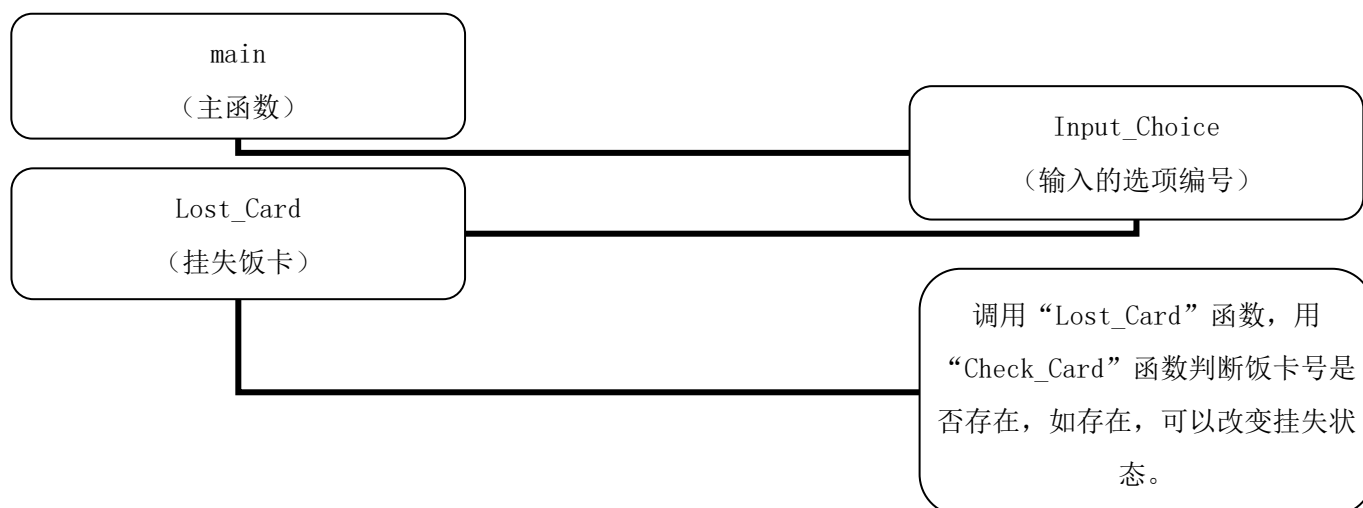
7. Charge_Money



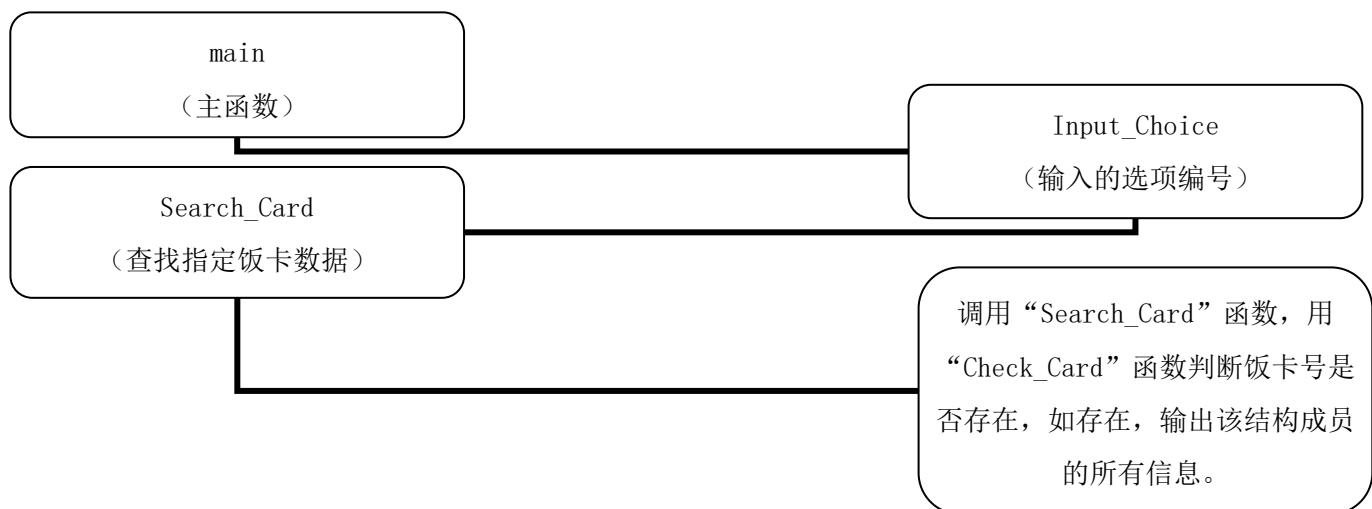
8. Delete_Card



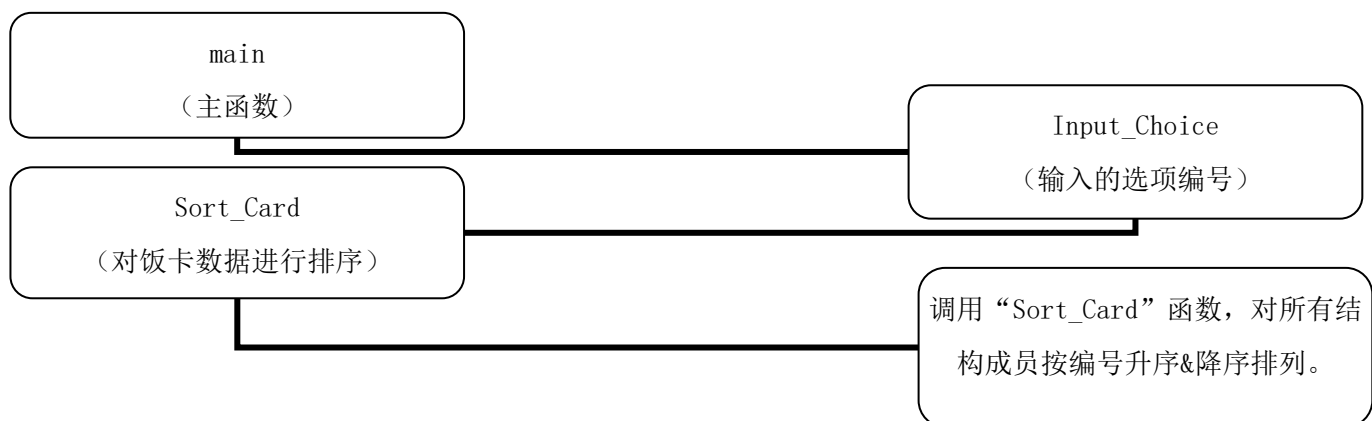
9. Lost_Card



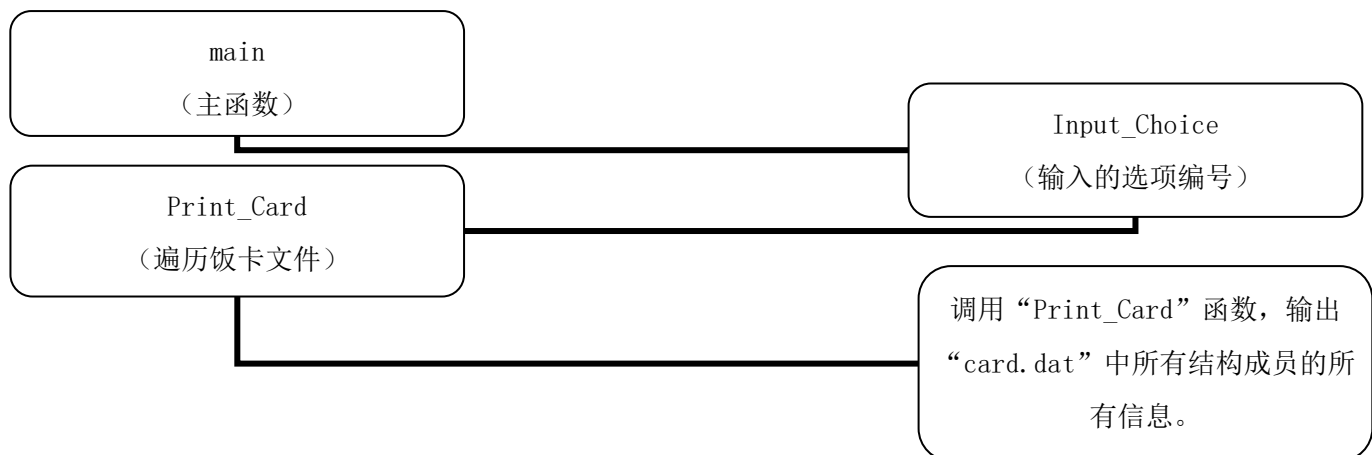
10. Search_Card



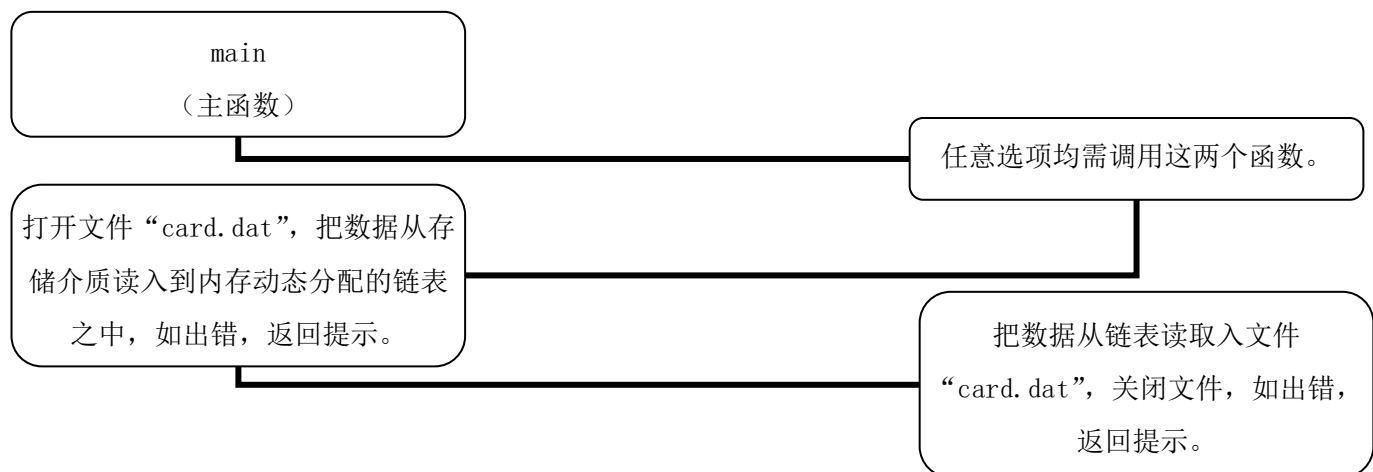
11. Sort_Card



12. Print_Card



13. Open_File&Close_File



3. 数据结构

```
#define C card_imf
#define S struct
#define I int
S C
{
    long number;
    char name[20];
    double money;
    I lost;
    S C *next;
};
```

在这个数据结构中，以 long 型定义的 number 表示饭卡号，char 型定义的 name 代表学生的姓名，double 型 money 代表饭卡中的余额，int 型 lost 代表饭卡是否挂失。

4. 程序代码设计

1. Input_Choice

用 for 循环+if\else 语句把可视化框架打出来，接收用户的返回值赋给主函数中的 switch 语句；

2. Build_File

从无到有新建一个“card.dat”文件。

3. Creat_new

从内存上动态分配若干个长度为该结构数组的空间作为链表，把存储介质上“card.dat”文件中的数据导入链表。

4. Start_Card

初次运行程序时输入的初始数据，由于不存在数据赋给头指针，若直接运行函数“Insert_Card”会造成错误，即群龙无首，所以增加“Start_Card”函数避免此问题。

5. Insert_Card

在通过“Start_Card”函数创建完成第一个结构数组的数据后，通过此函数创建更多的学生数据。

6. Use_Card

首先这个函数会调用“Check_Card”函数扫描一遍学生数据，判定该饭卡信息是否存在；如果不存在输出“饭卡不存在！”。如果存在，系统自动从该人的饭卡中减去饭钱，并分别显示买饭前后的金额，如果原来饭卡中的余额不足 5 元，或余额低于当前消费额，则不能买饭，并显示“余额不够，请续钱！”，以及如果挂失信息为真，则亦不能消费，并显示“本卡已冻结！”。

7. Charge_Money

首先这个函数会调用“Check_Card”函数扫描一遍学生数据，判定该饭卡信息是否存在；确定存在后会让用户输入充值额，充值后会显示充值前后的余额。以及如果挂失信息为真，则不能充值，并显示“本卡已冻结！”。

8. Delete_Card

首先这个函数会调用“Check_Card”函数扫描一遍学生数据，判定该饭卡信息是否存在；确定存在后使该行信息脱离链表，即删除该饭卡的信息，输出“旧饭卡注销成功！”。

9. Lost_Card

首先这个函数会调用“Check_Card”函数扫描一遍学生数据，判定该饭卡信息是否存在；

确定存在后再输出此卡的挂失状态，由用户变更此卡的挂失状态。

10. Search_Card

首先这个函数会调用“Check_Card”函数扫描一遍学生数据，判定该饭卡信息是否存在；确定存在后输出该饭卡信息。

11. Sort_Card

由用户决定将所有信息升序&降序排列，程序将用“选择排序法”对所有链表信息进行排序，排完后将链表中的数据输出到“card.dat”文件。

12. Print_Card

遍历饭卡文件，输出到屏幕。

13. Open_File&Close_File

各函数打开“card.dat”文件的方式不同，由各函数传回的 openmode 决定以何种方式打开文件；开关文件若不成功，输出相应的提示语。

5. 测试与调试

测试数据类型	测试数据合法性	测试数据	预期结果	实测结果
Input_Choice	是	4	跳转至 Insert_Card 或 Start_Card 函数	跳转成功，下一步输入饭卡的详细信息
Input_Choice	否	10	清屏	清屏
Use_Card	是	任意数字（此时测试数据组饭卡余额为 4）	无法消费并弹出“余额不足，请充值！”字样	与预期结果一致
Sort_Card	是	1	所有饭卡数据降序排列，输出“已按照您的选项进行排列！”字样	通过遍历函数输出后证明与预期结果一致
Sort_Card	否	3	输出“请输入正确的选项编号”字样	与预期结果一致
Search_Card	否	（输入一个文件中不存在的饭卡编号）	输出“饭卡不存在！”字样	与预期结果一致

第3部分 实训总结

1. 个人总结

(1) 通过宏定义有效减少了重复的定义名的输入，加快了编程效率；

(2) 由于平时上课有认真听，课后也有更加深入地学习相关的编程知识，所以一路写下来基本没有遇到困难，唯一遇到的「群龙无首」问题也很快通过 if-else 语句解决了。

(3) 本来想在遍历函数前写一个管理员身份验证的函数，但考虑到并没有什么卵用，毕竟“card.dat”整体未被加密，想看随时打开 dat 文件就能看了。再写一个验证步骤显得特别矫情。

(4) 这里详细为大家讲解一下如何通过两个指针就能实现链表的冒泡排序：首先与对数组进行冒泡排序的思路一致，各相邻结构数组比较某个成员的大小，不符合当前大小排序条件的就两两互换；以按照饭卡号升序（由小到大）排列为例，整个链表中饭卡号最大的那个结构数组通过这样的交换会成为整个链表的最后成员，这样还剩余（所有成员数-1）个成员需要排列，这样使用双 for 语句嵌套循环即可完成对整个链表的排序。而指针扮演的是搬运工的角色：用 p1、p2 分别指向满足交换条件相邻的两个成员，然后把前一个成员的值赋给过渡成员 temp，再让 temp（前一个成员的值）的 next 指向后一个成员的 next，再把后一个成员的值赋给前一个成员，把 temp（前一个成员的值）的值赋给后一个成员，此时前后两个成员的值已经互换了，最后把前一个成员（值为之前的后一个成员）的 next 赋给后一个成员（值为之前的前一个成员）即可。这样 p1、p2 依然指向前后两个已经交换了位置的成员，相较于本人之前考虑直接断链再接时发生的 p1、p2 所指向的成员发生颠倒的情况，这种办法更为简洁有效。接下来 p1、p2 指向其对应的 next，对下一组相邻的成员再次执行以上的成员交换，直到双循环结束。

(5) 其实之前是想做出更好的可视化窗口的，但问了一圈各路编程大佬之后还是放弃了，他们异口同声地告诉我，现在学的 C 语言其实是在学核心算法，算法最重要，这些算法搞定了以后可以用其他比 C 更好的语言写可视化窗口。后来一想，其实道理确实是这样的，一开始我们可能做不出什么像模像样的软件，但这就是学习的必经之路，只有一步一个脚印扎扎实实把基础的编程思想学好了，以后才有能力去做那些复杂的工程。所以对于我个人而言，稍安勿躁，步步为营，把该学的学好才是重点，而不是急于求成地迫切想做出什么实质性的东西，就算做出来，也未必是最优的。

2. 结束语

感谢赵玉明老师的悉心教导，感谢张楠同学能一直坚持与我针对程序进行有效的讨论，感谢@高玩梁同学、@小甲鱼老师对程序优化方面提供的建议。

附录 A 程序清单

```

#include<stdio.h>
#include<stdlib.h>
#include<windows.h>
#include<string.h>

/*没错，用多个宏定义简化代码，这可是最新操作！*/
/*惊喜不惊喜，意外不意外？*/

#define C card_imf
#define S struct
#define SIZE sizeof(S C)
#define T count
#define H head
#define I int
#define P printf
#define N scanf

S C
{
    long number;
    char name[20];
    double money;
    I lost;
    S C *next;
};

I user_choice;
FILE *fp;
S C *Creat_new();//初次运行程序时新建链表

void Open_File(char *openmode);//打开文件

void Close_File();//关闭文件
void Build_File();//初次运行程序时新建数据文件
void Use_Card(S C *H,I T);//消费
void Insert_Card(S C *H,I T);//插入新饭卡信息
void Charge_Money(S C *H,I T);//充值
void Delete_Card(S C *H,I T);//注销饭卡
void Lost_Card(S C *H,I T);//挂失饭卡
void Search_Card(S C *H,I T);//查找指定饭卡数据
void Start_Card(S C *H,I T);//初次运行程序时输入的初始数据
void Sort_Card(S C *H,I T);//对饭卡数据进行排序
void Print_Card(S C *H);//遍历饭卡文件
I Input_Choice();//输入的选项编号
I Check_Card(S C *H,I T,I num);//检查某编号的饭卡是否存在

I main()
{
    long num;

    while((user_choice=Input_Choice())!=0)
    {
        I T=1;
        S C *H,*p;

        if(user_choice==1)
            Build_File();

        H=Creat_new();
        if(H==NULL)//由于初次执行程序时会出现群龙无首的错误，所以把「第一次接收数据」和「第 N 次接收数据」分类讨论（我没别的意思，别想歪了）
        {
            switch(user_choice)
            {
                case 2:
                case 3:
                case 4:
                    Start_Card(H,T);
                    break;
                case 5:
                case 6:
                case 7:
                    P(" 请输入饭卡号:");
                    N("%d",&num);
            }
        }
    }
}

```


	else if(i==10)	de))==NULL)	Close_File();
	P("★	{	return H;
7--查询饭卡信息	★");	P("File open	}
	else if(i==11)	error!\n");	
	P("★	exit(0);	void fp_file(S C *H,I
8--排序饭卡信息	★");	}	flag)//把内存链表中储存的文
	else if(i==12)	}	本内容写回到存储介质中的文
	P("★		件
9--遍历饭卡信息	★");	S C *Creat_new()	{
	else if(i==13)	{	S C *p;
	P("★	Open_File("r");	Open_File("w");
0--退出系统	★");	S C data;	p=H;
	P("\n");	S C *H,*tail,*p;	while(flag>0)
	}	H=tail=NULL;	{
	P("\n 请输入选项编		
号:");		while(fscanf(fp,"%ld%s%lf%d"	fprintf(fp,"%ld\t%s\t%f\t%d\n",
		,&data.number,data.name,&data	p->number,p->name,p->money,
N("%d",&user_choice);		.money,&data.lost)!=EOF)	p->lost);
	return user_choice;	{	p=p->next;
	}	p=(S C	flag--;
		*)malloc(SIZE);	}
	void Close_File()		Close_File();
	{	p->number=data.number;	}
	if(fclose(fp))		
	{	strcpy(p->name,data.name);	void Build_File()
	P("Can not close		{
the file!\n");		p->money=data.money;	Open_File("w");
	exit(0);	p->lost=data.lost;	P(" 文件建立成功 !
	}	p->next=NULL;	\n");
	}	if(H==NULL)	Close_File();
		H=p;	system("pause");
	void Open_File(char	else	system("cls");
*openmode)			}
	{	tail->next=p;	
		tail=p;	void Use_Card(S C *H,I
if((fp=fopen("card.dat",openmo		}	T)

15

fp_file(H,flag);		T)
}	new_data.lost=0;	{
system("pause");	p1=(S C	S C *p;
system("cls");	*)malloc(SIZE);	S C new_data;
}		P("请输入饭卡号: ");
	p1->number=new_data.number;	
void Insert_Card(S C *H,I		N("%ld",&new_data.number);
T)	strcpy(p1->name,new_data.name);	P("请输入姓名: ");
{	e);	
char a[5];		N("%s",new_data.name);
S C	p1->money=new_data.money;	P("请输入金额: ");
*p1,*p2,new_data;		
p2=H;	p1->lost=new_data.lost;	N("%lf",&new_data.money);
strcpy(a,"是");		new_data.lost=0;
while(strcmp(a,"是	while(p2->next!=NULL)	p=(S C
")==0)		*)malloc(SIZE);
{	p2=p2->next;	
P("请输入饭卡		p->number=new_data.number;
号: ");	p2->next=p1;	
	p2=p1;	strcpy(p->name,new_data.name);
N("%ld",&new_data.number););
	p2->next=NULL;	
if(Check_Card(H,T,new_data.number)==1)	T++;	p->money=new_data.money;
P("饭卡已	fp_file(H,T);	p->lost=new_data.lost;
存在! \n");	}	H=p;
else	P("是否继续添	H->next==NULL;
{	加饭卡号: ");	fp_file(H,T);
P("请输入	getchar();	system("pause");
姓名: ");	gets(a);	system("cls");
	}	}
N("%s",new_data.name);	system("pause");	
P("请输入	system("cls");	void Delete_Card(S C *H,I
金额: ");	}	T)
		{
N("%lf",&new_data.money);	void Start_Card(S C *H,I	long num;

```

I flag=T;                                T--;                                while(T>0)
P("请输入饭卡号:");                      }                                {
N("%ld",&num);                            loop:p=H;
S C *p1,*p2,*p;                          while(flag>1)                    if(p->number==num)
p2=p1=H;                                {                                {

if(Check_Card(H,T,num)==-1)              fprintf(fp,"%ld\t%s\t%f\t%d\n",    if(p->lost==0)
    P("饭卡不存在!\n");                  p->number,p->name,p->money,        {
\n");                                    p->lost);
    else                                p=p->next;                        P("当前饭卡未挂失! 是否挂
    {                                flag--;                            失:");
        Open_File("w");                }
        while(T>0)                      Close_File();                    N("%s",a);
        {                                P("旧饭卡注销
            成功! \n");                if(strcmp(a,"是")==0)
        }                                p->lost=1;
        if(p1->number==num)              system("pause");                  }
        {                                system("cls");                      else
        }                                if(p->lost==1)
        {                                {
            void Lost_Card(S C *H,I      {
            T)
        }                                {
        else                            I flag=T;
        {                                char a[5];
            long num;                    N("%s",a);
            S C *p;                      if(strcmp(a,"是")==0)
            p=H;                          p->lost=0;
            P("请输入饭卡号:");
            N("%ld",&num);
        }                                }
        goto
    loop;                                if(Check_Card(H,T,num)==-1)      }
    }                                P("饭卡不存在!\n");            p=p->next;
    p2=p1;                            \n");                            T--;
    else                                }
    {                                fp_file(H,flag);
        p1=p1->next;
    }

```

```

        P("操作成功！\n");
    }
    system("pause");
    system("cls");
}

void Search_Card(S C *H,I T)
{
    long num;
    S C *p;
    p=H;
    P("请输入饭卡号:");
    N("%ld",&num);

    if(Check_Card(H,T,num)==-1)
        P("饭卡不存在！\n");
    else
    {
        P("饭卡号\t姓名\t金额\t是否挂失\n");
        while(T>0)
        {
            if(p->number==num)
            {
                if(p->lost==0)
                {
                    P("%ld\t%s\t%.2f\t%s\n",p->number,p->name,p->money,"是");
                }
                else
                {
                    P("%ld\t%s\t%.2f\t%s\n",p->number,p->name,p->money,"否");
                }
            }
            else
            {
                P("1. 由大到小排列\n2. 由小到大排列\n请选择排列方式: \n");
                while(i!=1&&i!=2)
                {
                    N("%d",&i);
                    if(i==1)
                    {
                        for(j=1; j<T; j++)
                        {
                            if((p1->number)<(p2->number))
                            {
                                temp=*p1;
                                temp.next=p2->next;
                                *p1=*p2;
                                *p2=temp;
                                p1->next=p2;
                            }
                            p2=p2->next;
                        }
                    }
                    if(i==2)
                    {
                        for(j=1; j<T; j++)
                        {
                            if((p1->number)>(p2->number))
                            {
                                temp=*p1;
                                temp.next=p2->next;
                                *p1=*p2;
                                *p2=temp;
                                p1->next=p2;
                            }
                            p2=p2->next;
                        }
                    }
                }
            }
        }
    }
}

void Check_Card(S C *H,I T,I num)
{
    S C *p;
    p=H;
    while(T>0)
    {
        if(p->number==num)
        {
            T--;
            p=p->next;
        }
        else
        {
            return -1;
        }
    }
    return 1;
}

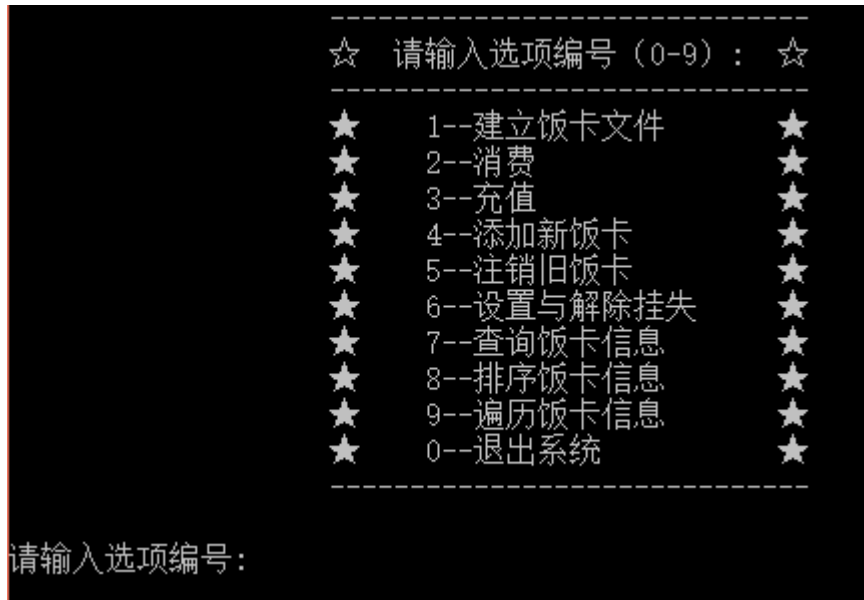
void Sort_Card(S C *H,I T)//通过两个指针就能实现链表的冒泡排序
{
    S C *p1,*p2,temp;
    I i,j,k;
}

```

p1=p1->next;	*p1=*p2;	
	}	void Print_Card(S C *H)
		{
p2=p1=H;	*p2=temp;	S C *p;
		P("学生饭卡信息如
	p1->next=p2;	下: \n");
}		P("饭卡号\t 姓名\t 金
else if(i==2)		额\t 是否挂失\n");
{	p2=p2->next;	
for(j=1; j<T;		for(p=H;p!=NULL;p=p->next)
j++)	p1=p1->next;	{
		if(p->lost==0)
p2=p2->next;	p2=p1=H;	P("%ld\t%s\t%.2f\t%s\n",p->nu
	}	mber,p->name,p->money,"否");
for(k=0; k<(T-j); k++)	}	
	else	else
		P("请输入
if((p1->number)>(p2->number)	正确的选项编号! \n");	P("%ld\t%s\t%.2f\t%s\n",p->nu
)	}	mber,p->name,p->money,"是");
	{	}
	P("已按照您的选项	system("pause");
	进行排列! \n");	system("cls");
temp=*p1;	fp_file(H,T);	
	system("pause");	}
temp.next=p2->next;	system("cls");	
	}	

附录 B 用户使用说明书

输入相应序号就能进入相关操作了，是不是很简单？



参考文献

- [1] 何钦铭. C 语言程序设计[M]. 2008 年 1 月第一版. 北京:高等教育出版社, 249-307

《C 程序设计综合实践》评分表

评 分 项 目	何承翰
完成情况（40%）	
答辩情况（20%）	
专业技能训练报告（20%）	
考勤与表现（20%）	
综合评分	

指导教师签名：_____

日 期： 2017 年 月 日