

中國石油大學勝利學院

# C 语言课程设计报告

题    目： 学生管理系统

学    号： 201506014217

学生姓名： 鲁超顺

系    别： 信息技术系

专业班级： 2015 软件工程 2 班

2016 年 7 月 17 日

# 目录

第一章 设计目的.....	1
1.1 设计内容、目的与要求.....	1
1.1.1 设计内容.....	1
1.1.2 目的与要求.....	1
1.1.3 主要功能.....	1
1.2 解决的问题.....	1
第二章 总体设计.....	2
2.1 系统框图.....	2
2.2 主函数流程图.....	3
第三章 详细设计.....	4
3.1 数据结构.....	4
3.1.1 函数说明.....	4
3.1.2 结构体说明.....	4
3.2 子函数流程图.....	5
3.2.1 从文本提取信息.....	5
3.2.2 查询学生信息.....	7
3.2.3 修改学生信息.....	7
3.2.4 保存输入或修改的信息.....	8
第四章 调试与测试.....	9
4.1 功能模块测试.....	9
第五章 总结.....	31
指导教师评语及成绩.....	32

# 学生管理系统

## 第一章 设计目的

### 1.1 设计内容、目的与要求

#### 1.1.1 设计内容

设计一个学生基本信息，学生成绩数据信息构成内容可自行设计（如：学号、姓名、性别、宿舍号码、电话号码等等），学生管理数据类型定义为结构类型。

#### 1.1.2 目的与要求

（1）录入、查询、删除保存、修改保存、插入、读取学生信息的功能。各项功能均可按照文字提示操作，浅显易懂。

（2）学生信息和成绩数据保存在文件中，运行程序时将学生基本信息和成绩数据从文件读取到内存中,并自动计算出相应的综合成绩和实得学分。对学生信息插入、删除、修改学生数据后，可以将内存中修改过的学生信息重新保存至文件。

#### 1.1.3 主要功能

（1）能录入、修改和增减学生基本信息

（2）能够对学生信息和成绩进行查询

### 1.2 解决的问题

可以有效系统的管理学生基本信息，使得对学生基本信息和成绩信息的管理更加便利，提高工作效率。

## 第二章 总体设计

### 2.1 系统框图

学生管理系统主要分为录入学生信息、学生信息查询、删除学生信息、学生信息修改、输出学生信息、插入学生信息等 6 个模块。

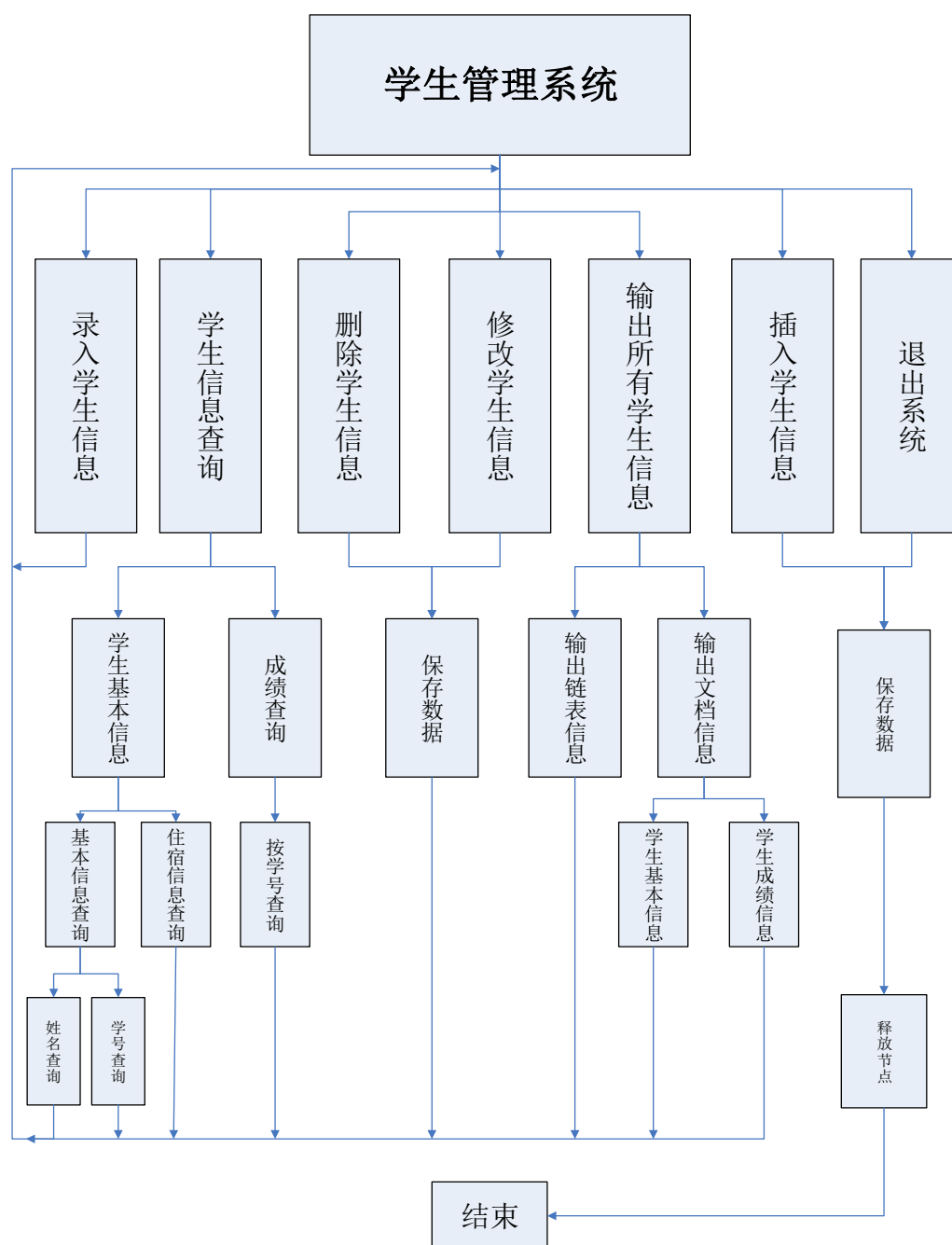


图 2-1 学生管理系统系统框图

## 2.2 主函数流程图

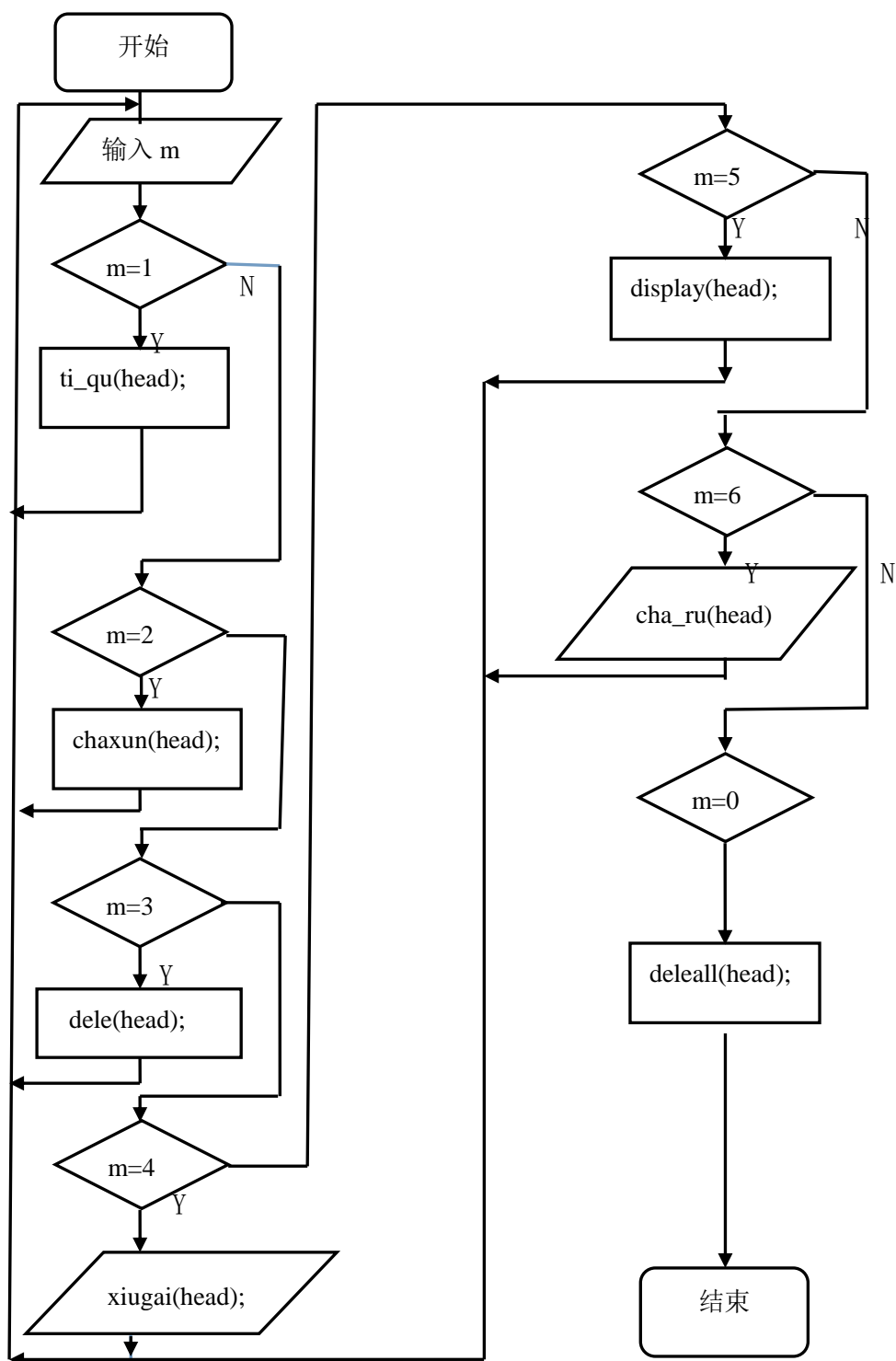


图 2-2,主函数的流程图

## 第三章 详细设计

### 3.1 数据结构

#### 3.1.1 函数说明

函数列表：程序中所用到函数说明。表结构如表 3-1 所示：

表 3-1 函数列表

函数名	函数类型	对应功能
ti_qu()	struct student	从 A 文本录入信息
tiqu()	struct student	从 B 文本录入成绩
chaxun()	struct student	查询学生信息
display()	void	输出全部学生信息
dele()	struct student	删除学生信息
deleall ()	struct student	释放链表全部结点
xiugai()	Void	修改某位学生信息
cha_ru()	struct student	插入学生信息
bao_cunA()	Void	保存信息到 A 文本
bao_cunB()	Void	保存信息到 B 文本

#### 3.1.2 结构体说明

表 3-2 学生信息列表 (struct student 表)

成员名	变量类型	对应中文名
xuehao[5]	char	学号
name[10]	char	姓名
xingbie[5]	char	性别

续表

成员名	变量类型	对应中文名
dianhua[11]	char	电话
kcID[10][10]	char	课程编号
kcname[10][10]	char	课程名称
xuefen	int	学分
chengji[10][10]	int	三类成绩
zonghe[10][2]	Double	最终成绩

## 3.2 子函数流程图

### 3.2.1 tiqu()//从文本提取信息

核心代码

```
while(!feof(fp))
{
    if(i!=0)
    {
        if((p=(struct student*)malloc(sizeof(struct student)))==NULL)//创建节点
        {
            printf("内存错误！");
            exit(0);
        }
        if(studenthead==NULL)
            studenthead=p;
        else
            studentp->next=p;          //连接节点
        f=studentp;
        studentp=p;
        fscanf(fp,"%s%s%s%d%s",studentp->xuehao,studentp->name,studentp->xingb,&studentp->sushe,studentp->dianhua);
        tt(studentp);
        tiqu(studentp);
        studentp->next=NULL;
    }else
    {
        for(i=0;i<5;i++)
```

```

        fscanf(fp,"%s",h);
    }
}
f->next=NULL;
free(p);
printf("读取完毕！ \n");
fclose(fp);
return studenthead;
}

```

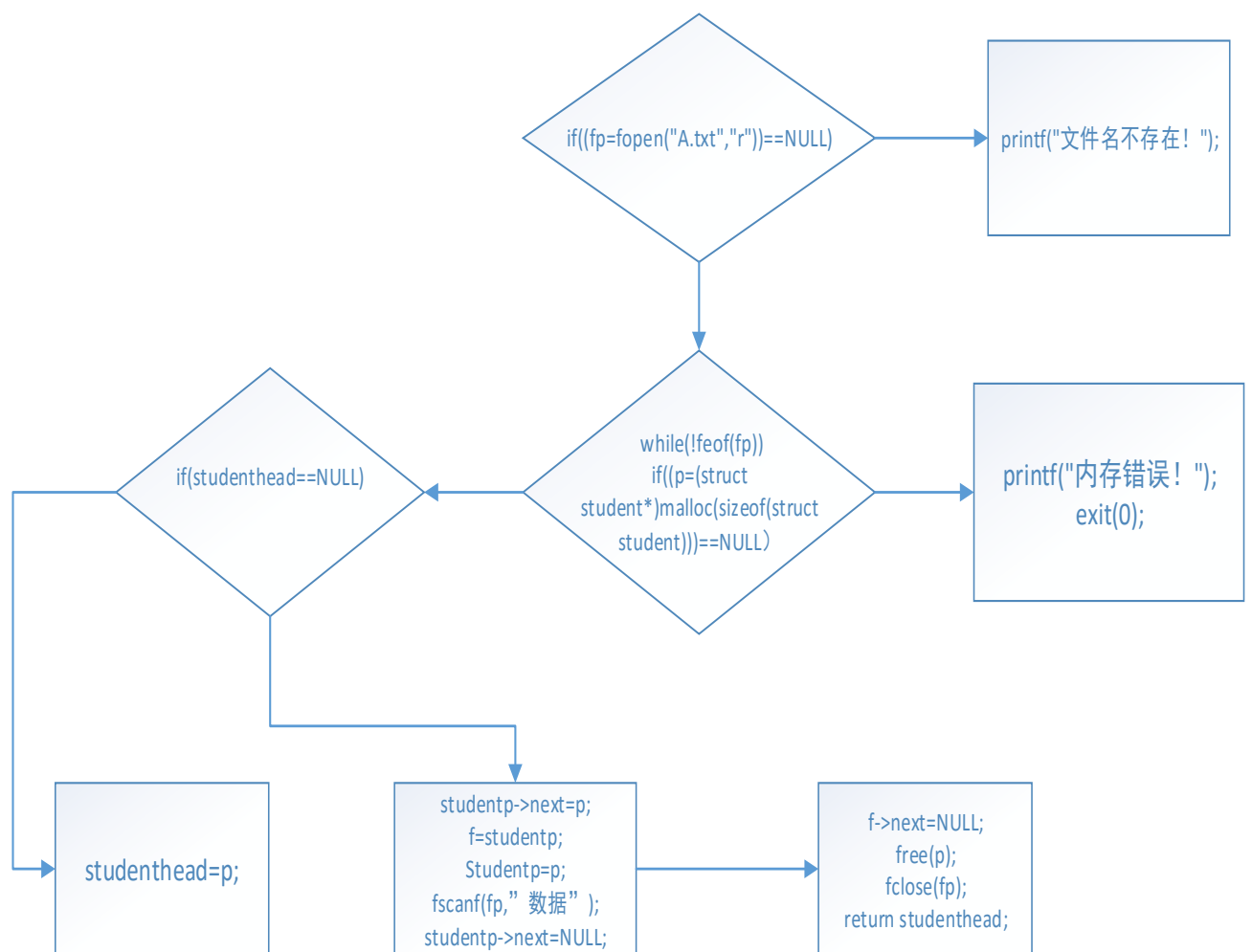


图 3-1 tiqu() 函数流程图



### 3.2.2 chaxun()//查询学生信息

首先输入 m 判断 m 的值,选择查询学生成绩还是学生基本信息通过指针 P 在链表中查找,最后输出。

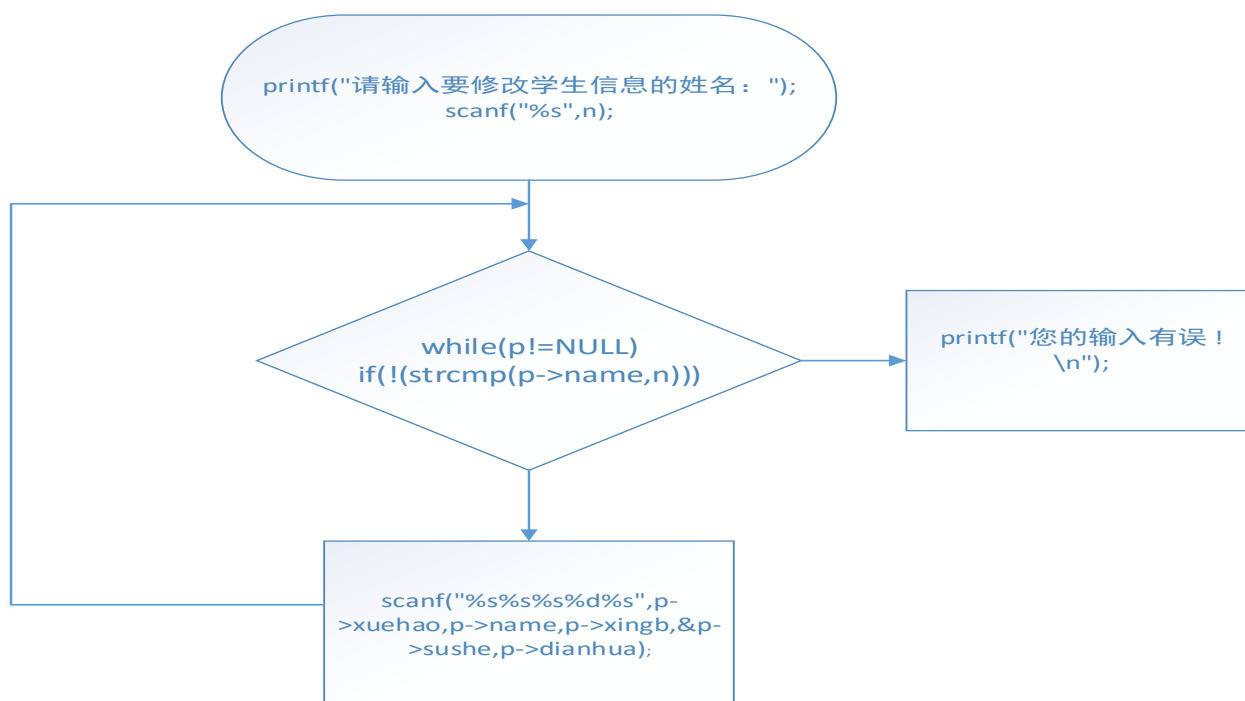


图 3-3chaxun()函数流程图

### 3.2.3 xiugai()//修改学生信息

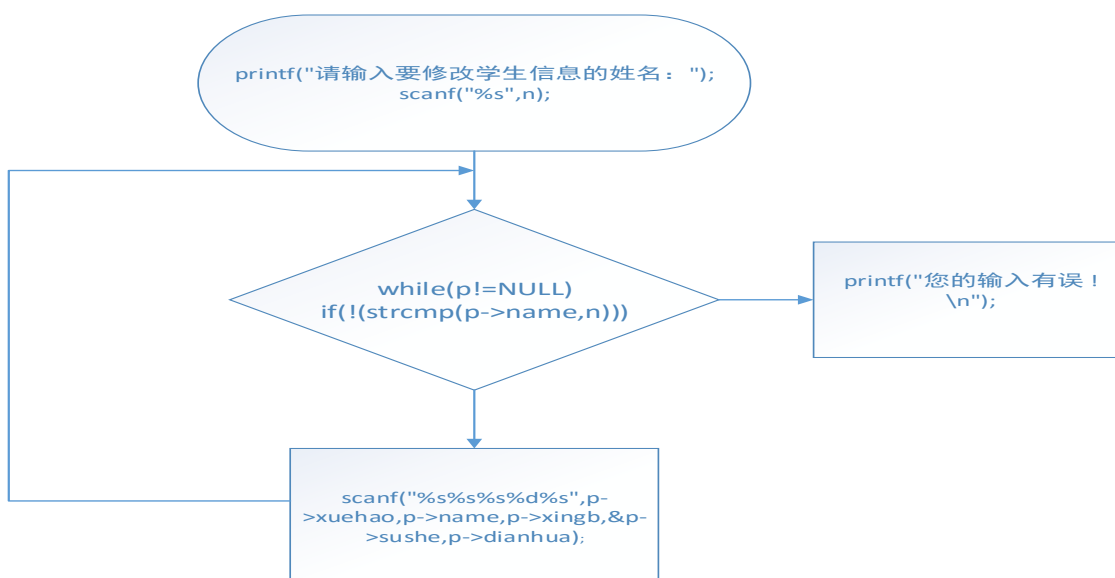


图 3-3xiugai()函数流程图

### 3.2.4 bao\_cunA() //保存输入或修改的信息

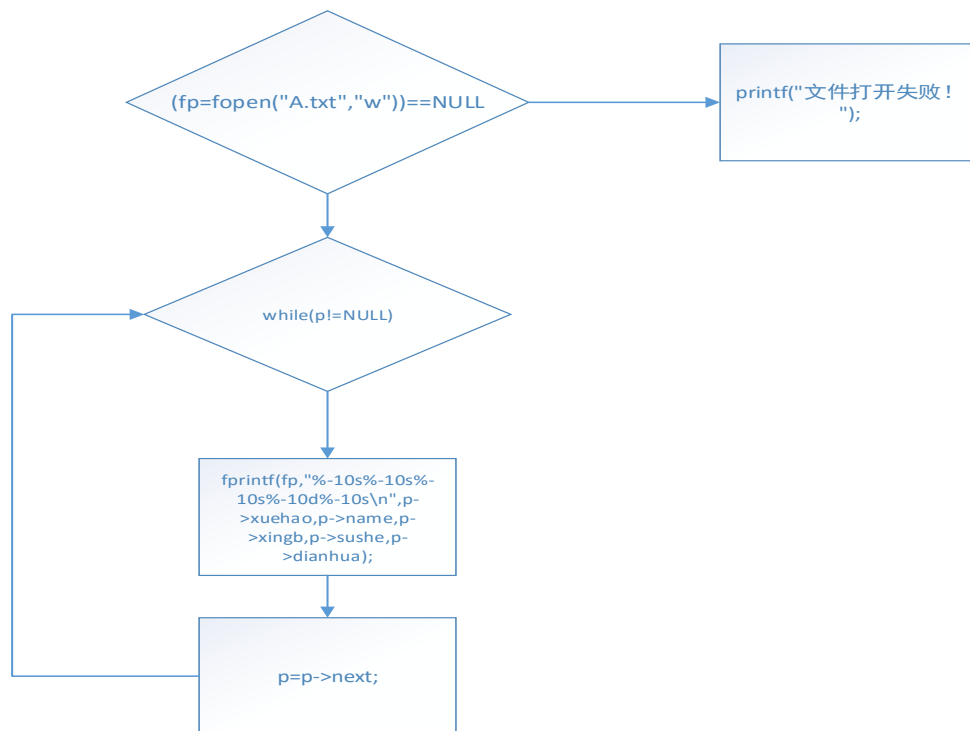
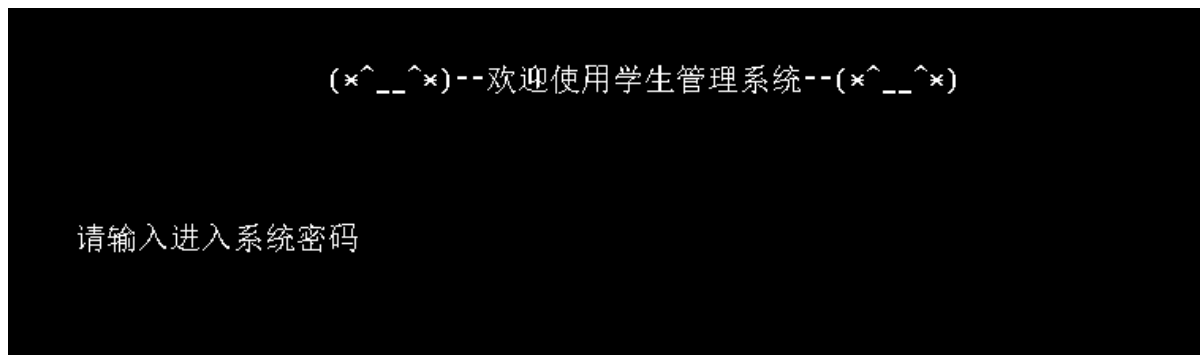


图 3-4bao\_cun 函数流程图

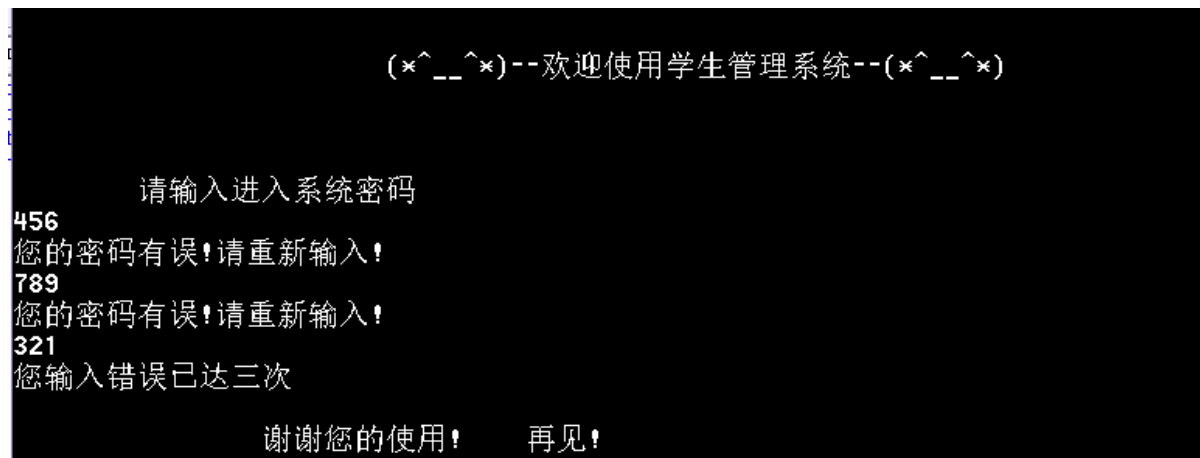
## 第四章 调试与测试

### 4.1 功能模块测试

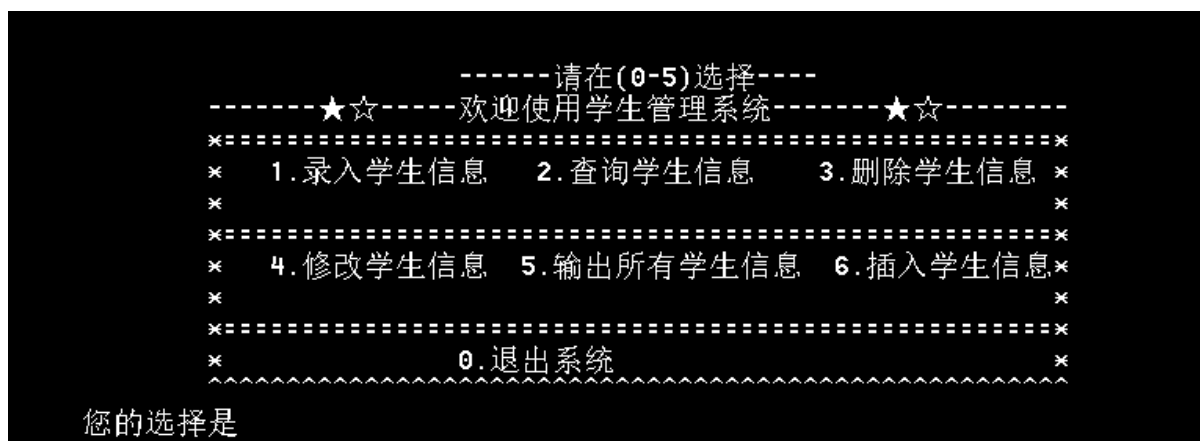
开始进入下界面



密码输入三次后退出系统



输入预设密码 123 进入主菜单界面



选择 1. 录入资料就有以下界面

```
读取完毕！
请按任意键继续. . .

          -----请在(0-5)选择-----
          -----☆☆-----欢迎使用学生管理系统-----☆☆-----
          ×=====×
          ×  1.录入学生信息   2.查询学生信息   3.删除学生信息  ×
          ×=====×
          ×  4.修改学生信息   5.输出所有学生信息  6.插入学生信息 ×
          ×=====×
          ×                      0.退出系统                      ×
          ×=====×
          ~~~~~

您的选择是 █
```

从键盘输入 2，输入 1 选择基本信息查询，再输入 1 按照姓名查询，输入：李果

```
          -----☆☆-----☆☆-----
          1.学生信息查询      2.成绩查询
          -----☆☆-----☆☆-----

1

          -----☆☆-----☆☆-----
          1.基本信息查询，2.住宿信息查询
          -----☆☆-----☆☆-----

1

          -----☆☆-----☆☆-----
          1.姓名查询          2.学号查询
          -----☆☆-----☆☆-----

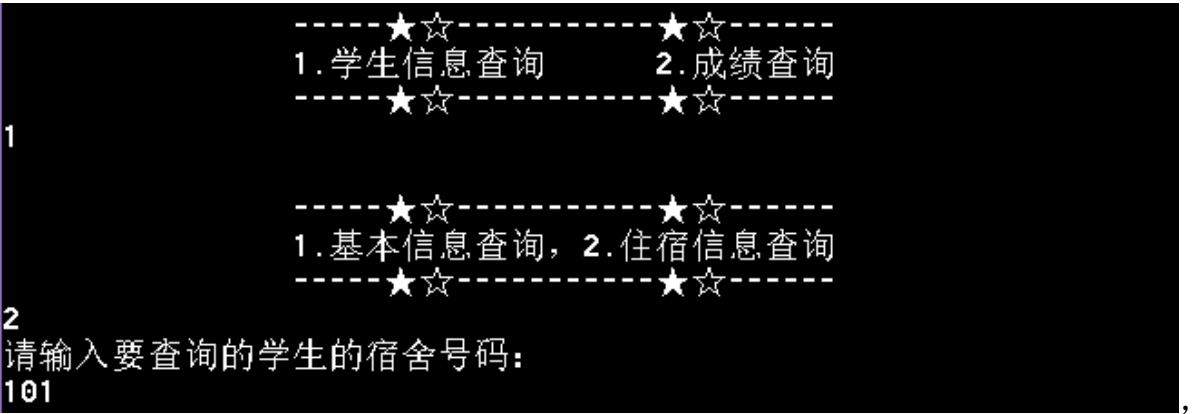
1
请输入要查询的学生的姓名：
```

输入学生姓名：李果

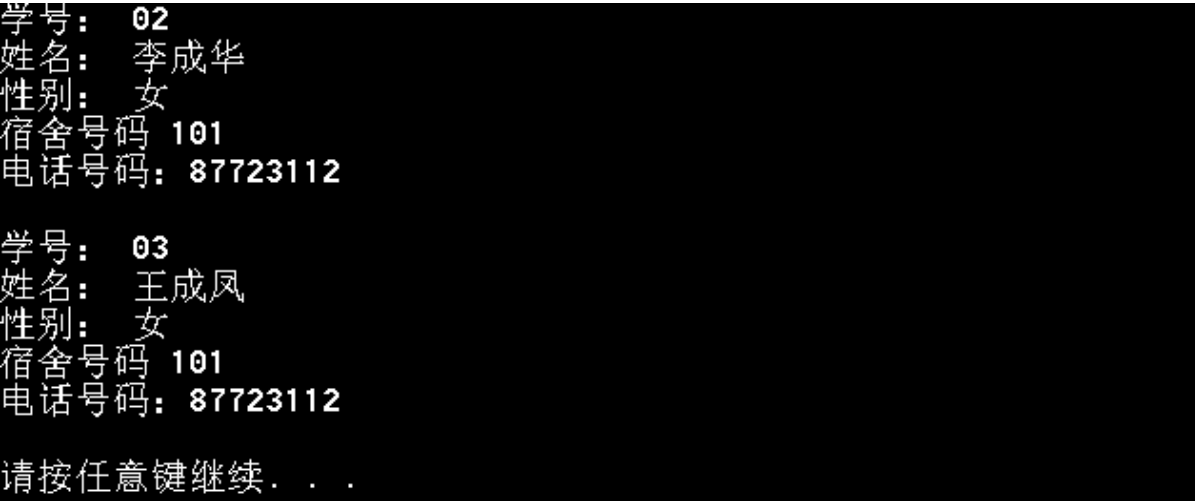
```
请输入要查询的学生的姓名：
李果

学号： 06
姓名： 李果
性别： 男
宿舍号码： 215
电话号码： 83445612
请按任意键继续. . .
```

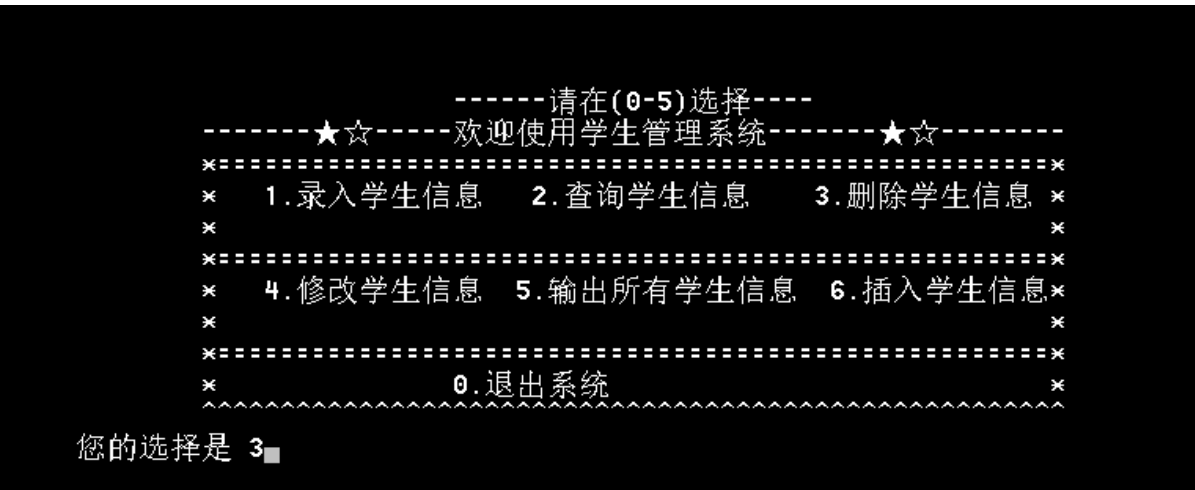
从主菜单界面输入 2，输入 1 选择基本信息查询，再输入 2 按照宿舍查询 输入 101



然后有如下界面



回到主菜单 选择 3



显示下一界面，输入要删除的学生的姓名



选择 1 保存后有如下提示，在菜单中选择 5 后

```
保存完毕！保存完毕！请按任意键继续...

          -----请在(0-5)选择-----
          -----★☆☆-----欢迎使用学生管理系统-----★☆☆-----
          ※=====※
          ※ 1.录入学生信息 2.查询学生信息 3.删除学生信息 ※
          ※=====※
          ※ 4.修改学生信息 5.输出所有学生信息 6.插入学生信息※
          ※=====※
          ※ 0.退出系统 ※
          ※~~~~~※

您的选择是 5■
```

可以看到李成华的信息已经被删除

```
请选择，1输出链表信息，2输出文档信息：
2
请选择 1.学生基本信息 2.学生成绩信息：
1
学号      姓名      性别      宿舍号码      电话
01      张成成      男      501      87732111
13      刘柳      女      101      79646132
03      王成凤      女      101      87723112
04      张明明      男      502      87734333
05      陈东      男      501      87732111
06      李果      男      215      83445612
17      刘华      男      210      61107446
32      赵军      男      215      7894566
07      张园园      女      102      87756122
    请按任意键继续... ■
```

回到主菜单 选择 4 修改张成成学生信息，并保存

```
请输入要修改学生信息的姓名：张成成
请对学生的学号、姓名、性别、宿舍号码、电话依次修改：
08 王天宇 男 501 110119112
是否保存修改：1.是 2.否
1
保存完毕！请按任意键继续... ■
```

回到菜单，查看学生信息，发现修改成功

```
请选择，1输出链表信息，2输出文档信息：
2
请选择 1.学生基本信息 2.学生成绩信息：
1
学号      姓名      性别      宿舍号码      电话
08        王天宇      男        501          110119112
13        刘柳        女        101          79646132
03        王成凤      女        101          87723112
04        张明明      男        502          87734333
05        陈东        男        501          87732111
06        李果        男        215          83445612
17        刘华        男        210          61107446
32        赵军        男        215          7894566
07        张园园      女        102          87756122
请按任意键继续. . .
```

在主菜单界面下，选择插入学生信息，输入 2

```
有1到10个位置可以选择，输入新插入位置：
2
```

输入相应的学生信息并且保存信息

```
有1到10个位置可以选择，输入新插入位置：
2
学号、姓名、性别、宿舍号码、电话号码：
02 李晨梓 男 501 65473421
是否保存： 1.保存      2.放弃
1
保存完毕！请按任意键继续. . .
```

查看信息可以看到 2 号位置已插入新的学生信息

```
请选择，1输出链表信息，2输出文档信息：
1
学号      姓名      性别      宿舍号码      电话
08        王天宇      男        501          110119112
02        李晨梓      男        501          65473421
13        刘柳        女        101          79646132
03        王成凤      女        101          87723112
04        张明明      男        502          87734333
05        陈东        男        501          87732111
06        李果        男        215          83445612
17        刘华        男        210          61107446
32        赵军        男        215          7894566
07        张园园      女        102          87756122
请按任意键继续. . .
```

回到主菜单选择 5

```
请选择，1输出链表信息，2输出文档信息：
2
请选择 1.学生基本信息 2.学生成绩信息：
2
学号    课程编号    课程名称    学分    平时成绩    实验成绩    卷面成绩    综合成绩    实得学分
01      B03      高等数学    4      71      -1      88      82.90      3.20
03      C56      英语        3      65      76      96      88.35      2.40
04      B03      高等数学    4      63      -1      80      74.90      3.00
05      B03      高等数学    4      45      -1      88      75.10      3.00
```

回到主菜单

```
-----请在(0-5)选择-----
-----★☆☆-----欢迎使用学生管理系统-----★☆☆-----
×=====×
× 1.录入学生信息 2.查询学生信息 3.删除学生信息 ×
×
×=====×
× 4.修改学生信息 5.输出所有学生信息 6.插入学生信息 ×
×
×=====×
×                      0.退出系统                      ×
×=====×
您的选择是 0
```

出下下面的界面，输入 1

```
是否保存修改，1保存      2不保存！
```

退出系统

```
感谢您的使用！      再见！

Press any key to continue█
```



## 附录：源程序

```
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
struct student
{
    char xuehao[5];        //学号
    char name[10];         //姓名
    char xingb[5];         //性别
    int  sushe;            //宿舍号
    char dianhua[11];      //电话
    char kcID[10][10];     //课程编号
    char kcname[10][10];  //课程名称
    int  xuefen[10];       //学分
    int  chengji[10][3];   //三门成绩
    double zonghe[10][2]; //最终成绩
    struct student *next;
};

void tt(struct student *p){
    int i=0;
    int j=0;
    for(i=0;i<10;i++)
        strcpy(p->kcID[i],"");
    for(i=0;i<10;i++)
        strcpy(p->kcname[10],"");
    for(i=0;i<10;i++)
        p->xuefen[i]=-1;
    for(i=0;i<10;i++)
        for(j=0;j<3;j++)
            p->chengji[i][j]=0;
    for(i=0;i<10;i++)
        for(j=0;j<2;j++)
            p->zonghe[i][j]=0;
}

struct student *ti_qu(struct student *head);           //从 A 文本录入基本信息
void tiqu(struct student *head);                       //从 B 文本录入基本信息
struct student *cha_ru(struct student *head);          //插入信息模块
void chaxun(struct student *head);                    //查询信息模块
void display(struct student *head);                   //输出模块
```



```

scanf("%d",&m);
system("cls");           //清除屏幕信息
switch(m)
{
case 1:head=ti_qu(head);break;
case 2:chaxun(head);break;
case 3:head=dele(head);break;
case 4:xiugai(head);break;
case 5:display(head);break;
case 6:head=cha_ru(head);break;
case 0:
    printf("是否保存修改, 1 保存\t2 不保存! ");
    scanf("%d",&d);
    if(d==1)
    { bao_cunA(head);
      bao_cunB(head);}
    head=deleall(head);break;      //退出
default:printf("\t 输入有误\n");
}
if(m==0) break;
system("pause");

}

system("cls");
printf("\n\n\n\n\t 感谢您的使用!\t 再见!\n\n\n");

}

```

struct student \*ti\_qu(struct student \*studenthead) //从 A.txt 文档中提取信息     studenthead 为传入的链表的头结点

```

{
    FILE *fp;           //定义文件指针
    struct student *p,*studentp,*f; // 定义结构体指针 p 为新结点指针, studentp 为链表操作的当前指针
    char h[20];
    int i=0;
    if((fp=fopen("A.txt","r"))==NULL) //以只读形式打开 A.txt 文件夹判断是否成功打开
    {
        printf("文件名不存在! ");
        exit(0);
    }
    studentp=studenthead;

```

```

while(!feof(fp))
{
    if(i!=0)
    {
        if((p=(struct student*)malloc(sizeof(struct student)))==NULL)//创建节点开辟空间
        {
            printf("内存错误！ ");
            exit(0);
        }
        if(studenthead==NULL)
            studenthead=p;
        else
            studentp->next=p;        //连接节点  p 为下一个结点的地址
        f=studentp;
        studentp=p;

        fscanf(fp,"%s%s%s%d%s",studentp->xuehao,studentp->name,studentp->xingb,&studentp->sushe,studentp->dianhua); //从文件中提取数据到程序中
        tt(studentp);
        tiqu(studentp);
        studentp->next=NULL;    //令下一个结点为空
    }else
    {
        for(i=0;i<5;i++)        //读取文件中的类型名称如：学号，性别，姓名.....
            fscanf(fp,"%s",h);    //从文件中提取数据给 h
    }
}
f->next=NULL;
free(p);
printf("读取完毕！ \n");
fclose(fp);    //关闭文件
return studenthead;
}

```

void tiqu(struct student \*p)/////从 B.txt 文档中提取信息

```

{
    FILE *fp;
    char h[20];
    int i=0;
    int y=0;
    char xuehao[10];
    char kcID[10];        //课程编号
}

```

```

char kcID1[10]="NULL"; //课程编号
char kcname[10];        //课程名称
int xuefen;              //学分
int chengji[3];          //三门成绩
double zonghe[2];        //最终成绩
double sum=0;
if((fp=fopen("B.txt","r"))==NULL)
{
    printf("文件名不存在！");
    exit(0);
}
while(!feof(fp))        //光标是否到文件的尾
{
    if(i!=0)
    {

        fscanf(fp,"%s%s%s%d%d%d",xuehao,kcID,kcname,&xuefen,&chengji[0],&chengji[1],&chengji
[2]); //从文件中提取数据到程序中

        if(!strcmp(xuehao,p->xuehao)&&strcmp(kcID,kcID1))
        {
            strcpy(kcID1,kcID);
            zonghe[0]=0;
            if(chengji[1]==-1)
            {
                sum=0.3*(chengji[0])+0.7*(chengji[2]);          //根据数据计算综合成
绩并存入数组内
                (zonghe[0])=sum;
                sum=0;
            }
            else {
                sum=0.15*(chengji[0])+0.15*(chengji[1])+0.7*(chengji[2]);
                zonghe[0]=sum;          //把计算出来的
综合成绩 sum 存到
                sum=0;
            }
            zonghe[1]=0;
            if((zonghe[0])>=90&&zonghe[0]<=100)          //根据综合成绩进一步计算实
得学分并存入数组内
                zonghe[1]=xuefen;
            else if((zonghe[0])>=80)
                zonghe[1]=xuefen*0.8;
            else if(zonghe[0]>=70)

```

```

        zonghe[1]=xuefen*0.75;
    else if(zonghe[0]>=60)
        zonghe[1]=xuefen*0.6;
    else
        zonghe[1]=0;

    strcpy(p->kcID[y],kcID);
    strcpy(p->kcname[y],kcname);
    p->xuefen[y]=xuefen;
    p->chengji[y][0]=chengji[0];
    p->chengji[y][1]=chengji[1];
    p->chengji[y][2]=chengji[2];
    p->zonghe[y][0]=zonghe[0];
    p->zonghe[y][1]=zonghe[1];
    y++;
    }
}
else
{
    for(i=0;i<7;i++)
        fscanf(fp,"%s",h);    //只会执行一遍，读取文件中的类型名称如：学号，课程名
称，课程编号.....
    }
}

fclose(fp);

}

```

```

void display(struct student *studenthead)    //-----输出所有信息
{
    FILE*fp;
    char ch;
    int i=0,j=0;
    struct student *p;    //定义结构体指针
    printf("请选择， 1 输出链表信息， 2 输出文档信息： \n");
    scanf("%d",&i);
    if(i==1)
    {
        if(studenthead==NULL)    //判断链表是否为空
        {
            printf("找不到学生信息！ \n");
        }
    }
}

```

```

        return;
    }
    p=studenthead;                //把链表表头复制指针 p，用 P 对链表进行操作
    printf("学号      姓名      性别      宿舍号码      电话  \n");
    while(p!=NULL)                //判断链表表头是否为空
    {

        printf("%-10s%-10s%-10s%-10d%-10s\n",p->xuehao,p->name,p->xingb,p->sushe,p->dianhua);
        p=p->next;                //p 指向下一结点
    }
}
else if(i==2)
{
    printf("请选择  1.学生基本信息  2.学生成绩信息: \n");
    scanf("%d",&j);
    if(j==1)
    {
        if((fp=fopen("A.txt","r"))==NULL)                //判断链表是否为空
        {
            printf("文件名不存在，找不到学生信息! ");        //提示错误信息
            exit(0);}
        while(!feof(fp))                //光标是否到文件的尾
        {
            ch=fgetc(fp);
            putchar(ch);                //输出文件信息
        }
        fclose(fp);                //关闭文件
    }
    else if(j==2)
    {
        if((fp=fopen("C.txt","r"))==NULL)
        {
            printf("文件名不存在，找不到学生信息! ");
            exit(0);}
        while(!feof(fp))
        {
            ch=fgetc(fp);
            putchar(ch);}
        fclose(fp);                // 关闭文件
    }
}
}

///-----查询

```

```

void chaxun(struct student *head)
{

    struct student *p;                //定义结构体指针 p
    char a[10],n[5];
    int m,k,l=0,x,j=0;                //定义标志 1 并赋值为 0
    p=head;                           //把链表表头复制指针 p，用 P 对链表进行操作
    if(head==NULL)                    //判断链表表头是否为空
    {
        printf("\t\t 学生信息为空；请先录入学生信息！ \n\n");
        return ;
    }
    printf("\n\t\t-----★☆☆-----★☆☆-----\n");
    printf("\t\t1.学生信息查询      2.成绩查询\n");
    printf("\t\t-----★☆☆-----★☆☆-----\n");
    scanf("%d",&m);
    if(m==1)
    {
        m=0;
        printf("\n\t\t-----★☆☆-----★☆☆-----\n");
        printf("\t\t1.基本信息查询, 2.住宿信息查询\n");
        printf("\t\t-----★☆☆-----★☆☆-----\n");
        scanf("%d",&k);
        if(k==1)
        {
            printf("\n\t\t-----★☆☆-----★☆☆-----\n");
            printf("\t\t1.姓名查询      2.学号查询   \n");
            printf("\t\t-----★☆☆-----★☆☆-----\n");
            scanf("%d",&m);
            if(m==1)
            {
                printf("请输入要查询的学生的姓名： \n");
                scanf("%s",a);
                while(p!=NULL)
                {
                    if(!strcmp(p->name,a))                //判断输入的名字和指针 p 指向
                                                            的名字是否相等，相等并返回 0 值
                    {
                        printf("\n 学号： %s\n 姓名： %s\n 性别： %s\n 宿舍号码： %d\n 电话
号码： %s\n",p->xuehao,p->name,p->xingb,p->sushe,p->dianhua);
                        l=1;                                //把 l 赋值为 1；
                    }
                    p=p->next;                               //p 指向下一结点
                }
            }
        }
    }
}

```



```

    }
    if(l==0) //如果上一判断执行，就不会有查找失败的提示。
    {
        printf("查找失败!!\n");
        l=0; //此判断执行，又把 l 归为零；
    }
}
else if(m==2)
{
    printf("请输入要查询的学生的学号： \n");
    scanf("%s",n);
    while(p!=NULL)
    {
        if(!strcmp(p->xuehao,n)) // 判断输入的学号和指针 p 指向的学号是否相等，相等并返回 0 值
        {
            printf("学号： %s\n 姓名： %s\n 性别： %s\n 宿舍号码： %d\n 电话号码： %s\n",p->xuehao,p->name,p->xingb,p->sushe,p->dianhua);
            l=1; //把 l 赋值为 1；
        }
        p=p->next; //p 指向下一结点
    }
    if(l==0) //如果上一判断执行，就不会有查找失败的提示。
    {
        printf("查找失败!!\n");
        l=0; //此判断执行，又把 l 归为零；
    }
}
else if(k==2)
{
    printf("请输入要查询的学生的宿舍号码： \n");
    scanf("%d",&x);
    system("cls"); //清除屏幕信息
    while(p!=NULL)
    {
        if(p->sushe==x) // 判断输入的宿舍号码和指针 p 指向的宿舍号码是否相等
        {
            printf("学号： %s\n 姓名： %s\n 性别： %s\n 宿舍号码 %d\n 电话号码： %s\n",p->xuehao,p->name,p->xingb,p->sushe,p->dianhua);l=1;

```

```

        }
        p=p->next;                //p 指向下一结点
    }
    if(l==0)
    {
        printf("查找失败!!\n");
        l=0;
    }

}

}
else if(m==2)
{
    struct student *p;
    char n[5];
    p=head;
    printf("请输入要查询的学生的学号: \n");
    scanf("%s",n);
    system("cls");                //清除屏幕信息
    while(p!=NULL)
    {
        if(!(strcmp(p->xuehao,n)))    // 判断输入的学号和指针 p 指向的学号是否相等
相等返回 0 值
        {
            printf("学号%5s\t 姓名:%5s\n",p->xuehao,p->name);
            for(j=0;j<10;j++)
                if(p->xuefen[j]!=-1)
                    printf("学分%3d 课程名称:%10s\t 课程编号:%10s\t 综合成绩:%5.2lf\t 实
得学分:%5.2lf\n",p->xuefen[j],p->kcname[j],p->kcid[j],p->zonghe[j][0],p->zonghe[j][1]);l=1;
        }
        p=p->next;
    }
    if(l==0)
    {
        printf("查找失败!!");
        l=0;
    }
}

}

```

```

//-----删除学生资料
struct student *dele(struct student *studenthead)
{
    struct student *p,*studentp,*R;           // 定义结构体指针
    int k,l=0,j;
    char n[10];                                //定义一个字符串数组
    if(studenthead==NULL)                      //判断表头是否为空
    {
        printf("学生信息不存在，请录入学生信息！\n");
        return studenthead;                   //链表表头为空 返回头指针
    }
    printf("输入要删除的学生的姓名：\n");
    scanf("%s",n);k=1;
    R=studenthead;                             //把链表表头复制指针 R，用 R 对链表进行操作
    while(R!=NULL&&(strcmp(n,R->name)))        // 判断输入的姓名和指针 R 指向的姓名是否
    相等 相等返回 0 值
    {
        studentp=R;
        R=R->next;
        k=k+1;
        l=1;
    }
    if(R==NULL||(strcmp(n,R->name)))
        printf("删除无效!\n");
    else if(k==1)
    {
        p=studenthead;                        //p 指向第 1 个结点
        studenthead=p->next;                  //第 1 个结点脱链，第二个结点变为第一个结点
        printf("删除成功!\n");
        printf("是否保存:  1.保存  \t2.放弃\n");
        scanf("%d",&j);
        if(j==1)
        {   bao_cunA(studenthead);   bao_cunB(studenthead);} //调用保存函数，对修改的信
    息保存
        free(p);                             //释放删除的节点
    }
    else if(R!=NULL&&!(strcmp(n,R->name)))
    {
        p=studentp->next;                    //把下一个节点赋值给 p
        studentp->next=p->next;              //把 p->next 节点赋值给 studentp->next
        printf("删除成功!");
        printf("是否保存:  1.保存  \t2.放弃\n");
        scanf("%d",&j);
    }
}

```

```

        if(j==1)
        {   bao_cunA(studenthead);   bao_cunB(studenthead);}
        free(p);
    }
    return studenthead;
}

//-----修改学生信息
void xiugai(struct student *head)
{
    struct student *p;           //定义结构体指针
    char n[10];int m;
    p=head;                      //把链表表头复制指针 p，用 p 对链表进行操作
    printf("请输入要修改学生信息的姓名：");
    scanf("%s",n);
    while(p!=NULL)               //判断指针是否为空
    {
        if(!strcmp(p->name,n))    // 判断输入的姓名和指针 p 指向的姓名是否相
        等 相等返回 0 值
        {
            printf("请对学生的学号、姓名、性别、宿舍号码、电话依次修改：\n");    //依
            次修改学生信息
            scanf("%s%s%s%d%s",p->xuehao,p->name,p->xingb,&p->sushe,p->dianhua);
            printf("是否保存修改：1.是\t2.否\n");    //提示是否保存
            scanf("%d",&m);
            if(m==1)
            {
                bao_cunA(head);    //调用 bao_cunA 函数 保存数据

            }
            else if(m==2)
            {
                main();            //不保存 返回主菜单
            }
            return ;
        }
        p=p->next;                //p 指向下一结点
    }
    printf("您的输入有误！\n");
    return ;
}

```

```

struct student *cha_ru(struct student *studenthead)    //-----插入学生资料
{
    struct student *p,*studentp;
    int k,i=0,j;
    if(studenthead==NULL)
    {
        printf("学生信息为空，无法插入!\n");    /*如果链表为空*/
        return studenthead;                    //返回链表表头
    }
    p=studenthead;                            //把链表表头复制指针 p，用 p 对链表进行操作
    while(p!=NULL)
    {
        p=p->next;                            //p 指向下一节点
        i++;                                //寻找有多少可以插入的节点
    }
    printf("有 1 到%d 个位置可以选择，输入新插入位置： ",i+1);
    scanf("%d",&i);
    k=1;
    studentp=studenthead;                    //把链表表头复制指针 studentp，用 studentp 对链表进行
操作
    while(studentp!=NULL&& k<i-1)
    {
        studentp=studentp->next;
        k=k+1;
    }
    if(studentp==NULL||i<1)
        printf("插入位置为无效值\n");
    else if(i==1)                            ///头节点插入
    {
        if((p=(struct student*)malloc(sizeof(struct student)))==NULL)
        {
            printf("内存申请错误！ \n");
            exit(0);
        }
        printf("学号、姓名、性别、宿舍号码、电话号码： \n");
        scanf("%s%s%s%d%s",p->xuehao,p->name,p->xingb,&p->sushe,p->dianhua);    ///插入
信息
        p->next=studenthead;
        studenthead=p;
        printf("是否保存:  1.保存  \t2.放弃\n");
        scanf("%d",&j);

```

```

        if(j==1)
            bao_cunA(studenthead);
    }
    else
    {
        if((p=(struct student*)malloc(sizeof(struct student)))==NULL)           //不是头节点
的插入
        {
            printf("内存申请错误！ \n");
            exit(0);
        }
        printf("学号、姓名、性别、宿舍号码、电话号码： \n");
        scanf("%s%s%s%d%s",p->xuehao,p->name,p->xingb,&p->sushe,p->dianhua);           ///插
入信息
        p->next=studentp->next;
        studentp->next=p;
        printf("是否保存： 1.保存 \t2.放弃\n");
        scanf("%d",&j);
        if(j==1)
            bao_cunA(studenthead);
    }

    return studenthead;
}

```

```

void bao_cunA(struct student *studenthead)    //-----将链表信息保存到 A.txt 文本文件
{
    struct student *p;
    FILE *fp;
    if((fp=fopen("A.txt","w"))==NULL)
    {
        printf("文件打开失败！ ");
        exit(0);
    }
    if(studenthead==NULL)
    {
        printf("打开失败，链表为空！ ");
        return;
    }
    p=studenthead;
    fprintf(fp,"学号      姓名      性别      宿舍号码      电话      \n");           //把链表表头复制指针 p，用 p 对链表进行操作

```

```

while(p!=NULL)
{

fprintf(fp,"%-10s%-10s%-10s%-10d%-10s\n",p->xuehao,p->name,p->xingb,p->sushe,p->dianhua);
    p=p->next;                //指向下一节点
}
fclose(fp);                //关闭文件
printf("保存完毕！ ");
return;
}

void bao_cunB(struct student *studenthead)    //----将链表信息保存到 B.txt 文本文件
{
    struct student *p;
    FILE *fp;
    int i=0;
    if((fp=fopen("C.txt","w"))==NULL)
    {
        printf("文件打开失败！ ");
        exit(0);
    }
    if(studenthead==NULL)
    {
        printf("打开失败，链表为空！ ");
        return;
    }
    p=studenthead;                //把链表表头复制指针 p，用 p 对链表进行
操作
    fprintf(fp,"学号    课程编号    课程名称    学分    平时成绩    实验成绩    卷面成绩\n");
    printf("综合成绩    实得学分    \n");
    while(p!=NULL)
    {
        for(i=0;i<10;i++)
            if(p->xuefen[i] != -1)

            fprintf(fp,"%-10s%-10s%-15s%-10d%-10d%-10d%-10d%-11.2lf%-10.2lf\n",p->xuehao,p->kcID[i],
p->kcname[i],p->xuefen[i],p->chengji[i][0],p->chengji[i][1],p->chengji[i][2],p->zonghe[i][0],p->zonghe[i][1]);

        p=p->next;
    }
    fclose(fp);
}

```

```

    printf("保存完毕！ ");
    return;
}

struct student *deleall(struct student *studenthead)/////-----释放节点
{
    struct student *p;                //定义结构体指针 p
    p=studenthead;                    //把链表表头复制指针 p, 用 p 对链表进行
操作
    while(p!=NULL)
    {
        studenthead=p->next;          //把 p->next 赋值给 studenthead , 然后释放 p 节点
        free(p);
        p=studenthead;
    }
    return p;
}

```



## 第五章 总结

通过这次的课程设计，增加了我对学习 C 语言学习的兴趣，我从 C 语言这门课程开始，已经发现程序设计的乐趣，在学习 C 语言的过程中也学到了许多计算机应用基础知识，对计算机的运行也有一个大体的了解。这次课程设计经过老师的讲解，自己的查找，在网上的搜索，实现题目上的要求，先做出一些简单的函数和程序，一步步的在做出其它的内容，不过在实际操作过程中犯得一些错误也还是会得到一些意外的收获。在具体的操作中，对这学期的 C 语言的理论知识得到巩固，在设计的过程中也发现自己的不足之处，在以后的学习和上机的操作中会更加注意，同时体会到 C 语言的简洁和灵活。在上机操作中我们的任何不懂的问题都会暴露出来，对于我们来说是件好事，在课程设计的过程中特别是对数组和循环有了深刻的理解。通过实际的操作，学会 C 语言程序编程的基本步骤、基本方法，开发了自己的逻辑思维能力，培养了分析问题，解决问题的能力。

