**《高级程序设计语言》**

**课程设计报告**

**学 院：**

**班 级：**

**学 号：**

**姓 名：**

1. **引言**（问题描述：题目要解决的问题是什么。）

从分析问题入手，设计可行的算法，进而编出相应的C程序。强调上机实践过程，从实际应用出发，编写学生成绩管理系统。  
 学会使用单链表，结构体及函数，独立设计编程实现以下功能:

1.输入学生信息  
2.删除学生信息  
3.增添学生信息

4.显示所有学生信息

5.按查姓名查找学生信息

6.计算总分和平均分

7.学生成绩排名

8.按查学号查找学生信息

2. **算法设计**

（1）总体思路规划：

计算数据

查找信息

输出信息

排序

修改信息

创建信息

学生管理系统

主函数

（circulate（）循环函数调用其他函数）

（2）创建信息思路：

输入学生总人数i个

用p1,head共同指向一段内存，用p2指向一段内存

输入数据，把p2,链接在p1的后面

执行i层循环后，返回主界面

stud\* Creat() //创建函数

{

system("cls");

printf("\t\t HELLO，您正在输入学生信息... \n\n");

printf("\t\t 请根据提示输入学生信息: \n\n");

int i,j;

stud \*head,\*p1,\*p2;

printf("\t\t\t ◎请输入总数:");

scanf("%d",&i);

if(i<=0)

{

printf("\t\t\t输入错误请重试\n");

system("pause");//暂停屏幕

}

if((head=(stud \*)malloc(sizeof(stud)))==NULL)//创建头指针

{

printf("\t\t\t空间不足建立失败\n");

exit(0);

}

head->next=NULL;

p1=head;

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

{

if((p2=(stud \*)malloc(sizeof(stud)))==NULL)//创建新的节点

{

printf("\t\t\t空间不足建立失败\n");

exit(0);

}

//输入学生信息

printf("\t\t您正在输入第%d个人的信息\n",j+1);

printf("\t\t ◎请输入学号:");

scanf("%s",p2->num);

printf("\t\t ◎请输入姓名:");

scanf("%s",p2->name);

printf("\t\t ◎请输入语文成绩:");

scanf("%d",&p2->chinese);

printf("\t\t ◎请输入数学成绩:");

scanf("%d",&p2->math);

printf("\t\t ◎请输入英语成绩:");

scanf("%d",&p2->English);

p2->sum=p2->chinese+p2->English+p2->math;

//使上一个节点指向下一个节点

p1->next=p2;

p2->next=NULL;//使最后节点next为空

p1=p2;

if(i>=3 && (j+1)%2==0) system("cls");

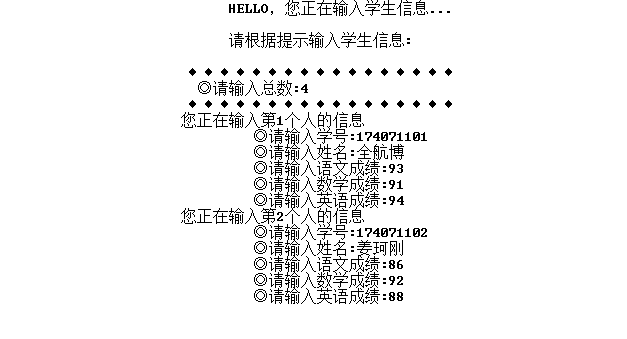
}

printf("\t\t已成功输入学生信息\n");

system("pause");

return (head);//返回头指针

}

（2）修改信息思路：

开辟内存空间

开辟内存空间

找到要插入(或删除)的数据位置

找到要插入(或删除)的数据位置

把要插入的数据地址保存在上个节点的 把删除节点的下个节点的地址覆盖

把要删除节点的下个节点的地址覆盖

删除节点之前的p->next中，并分情况（通过学号或者姓名）删除

除节点之前的p->next中；

把要插入的数据地址保存在上个节点的p->next中

删除节点之前的p->next中；

p->next中；

把当前插入数据的p->next用下个节点的地址赋值

用free()函数释放当前删除节点

把当前插入数据的p->next用下个节点的 用free()函数释放当前删除节点

判断y or n 是否继续删除

判断y or n 是否继续添加

插入（删除）成功后返回主页面

int Add(stud \*p)//添加函数

{

if(p==NULL || p->next==NULL)

{

printf("\t\t\t请先输入学生信息\n");

system("pause");

return 4;

}

stud \*q=p,\*a=p,\*head[100];

int i=-1,b,n=0;

char m='y';

while(q->next)

{

n++;

q=q->next;

}

while(m=='y')

{

system("cls");

q=a;

printf("\t\t\t您正在添加学生信息...:\n");

if((head[++i]=(stud \*)malloc(sizeof(stud)))==NULL)

{

printf("\t\t\t空间不足建立失败\n");

exit(0);

}

printf("\t\t\t需要插入到第b个数据之后(0<=b<=%d):",n);

scanf("%d",&b);

if(b<0 || b>n)

{

printf("\n\t\t输入错误\n");

system("pause");

break;

}

//插入到第b个数据之后

for(n=1;n<=b;n++)

q=q->next;

head[i]->next=NULL;

printf("\t\t◎请输入学号:");

scanf("%s",head[i]->num);

printf("\t\t◎请输入姓名:");

scanf("%s",head[i]->name);

printf("\t\t◎请输入语文成绩:");

scanf("%d",&head[i]->chinese);

printf("\t\t◎请输入数学成绩:");

scanf("%d",&head[i]->math);

printf("\t\t◎请输入英语成绩:");

scanf("%d",&head[i]->English);

head[i]->sum=head[i]->chinese+head[i]->English+head[i]->math;

head[i]->next=q->next;

q->next=head[i];

printf("\t\t\t输入成功\n");

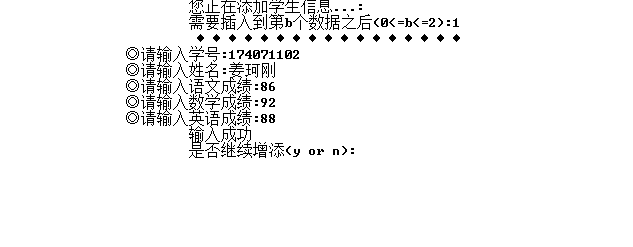
printf("\t\t\t是否继续增添(y or n):");

scanf(" %c",&m);

}

return 1;

}



int Del(stud \*p) //删除函数

{

if(p==NULL || p->next==NULL

{

printf("\t\t\t请先输入学生信息\n");

system("pause");

return 0;

}

stud \*q=p->next,\*a=p->next,\*h=p;//h为q的前一个节点

char m='y',b[20],n[20];

int j;

while(m=='y')

{

if(p==NULL)

{

printf("\t\t\t请先输入学生信息\n");

system("pause");

return 0;

}

system("cls");

q=a;

h=p;

printf("\t\t 您正在删除学生信息 ... \n");

printf("\t\t \n");

printf("\t\t 请选择删除方式 \n");

printf("\t\t \n");

printf("\t\t 1.按学号删除 \n");

printf("\t\t \n");

printf("\t\t 2.按姓名删除 \n");

printf("\t\t \n");

printf("\t\t 0.退出 \n");

printf("\n");

printf("\t\t 请输入您的选择:");

scanf("%d",&j);

switch(j)

{

case 1:

printf("\t\t\t请输入学号:");

scanf("%s",b);

while(q)

{

if(strcmp(q->num,b)==0)

{

h->next=q->next;//使前一个节点指向后一个

free(q);

printf("\t\t\t删除成功\n");

system("pause");

break;

}

h=q;//h为q的前一个节点

q=q->next;

}

if (q==NULL)

{

printf("\t\t\t无此学生\n");

system("pause");

break;

}

if(q)

{

printf("\t\t\t是否继续删除(y or n):");

scanf(" %c",&m);

}

break;

case 2:

printf("\t\t\t请输入姓名:");

scanf("%s",n);

while(q)

{

if(strcmp(q->name,n)==0)

{

h->next=q->next;//使前一个节点指向后一个

free(q);

printf("\t\t\t删除成功\n");

break;

}

h=q;

q=q->next;

}

if (q==NULL)

{

printf("\t\t\t无此学生\n");

system("pause");

break;

}

if(q)

{

printf("\t\t\t是否继续删除(y or n):");

scanf(" %c",&m);

}

break;

case 0:

m='n';

break;

default:

printf("\t\t\t输入错误\n");

system("pause");

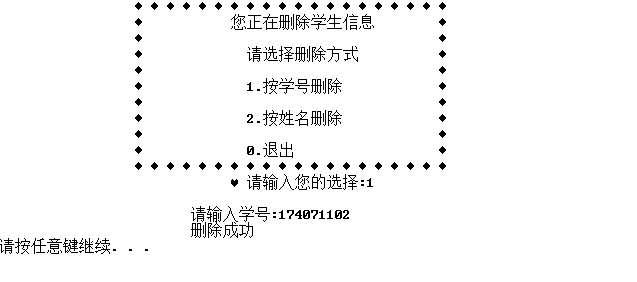
break;

}

}

return 1;

}



（3）查找信息思路：

判断当前所指向的数据是否和要查找的信息姓名（学号）一致

判断当前所指向的数据是否和要查找的信息姓名（学号）

不同则用p=p->next使之指向一个节点，继续执行上

不同则用p=p->next使之指向一个节点，继续执行上一个判断条件

直至p=NULL，如果没有找到则输出提示：没有找到；

直至p=NULL，分两种情况（学号查找，姓名查找）找到后并输出；找不到则输出提示：无此学生

找到了则输出

判断y or n 是否继续查找

返回主界面

int Search(stud \*p) //按姓名查询函数

{ //传头指针过来

if(p==NULL || p->next==NULL)//防止无学生信息

{

printf("\t\t\t请先输入学生信息\n");

system("pause");

return 0;

}

stud \*q=p->next,\*a=p->next;

char n[20],b[20];

char m = 'y';

while(m == 'y')//当m=‘y’时循环

{

q=a;//每次循环重置指针q

system("cls");

printf("\t\t\t请输入姓名:");

scanf("%s",b);

printf("\n");

while(q)

{

if(strcmp(q->name,b)==0)//比较节点中的姓名是否一致

{

printf("\t\t\t姓名:%s\n",q->name);

printf("\t\t\t学号:%s\n",q->num);

printf("\t\t\t语文成绩:%d\n",q->chinese);

printf("\t\t\t数学成绩:%d\n",q->math);

printf("\t\t\t英语成绩:%d\n",q->English);

break;

}

else q=q->next;

}

if(q)//等同于q!=NULL;

{

printf("\t\t\t是否继续查询(y or n):");

scanf(" %c",&m);

}

if (q==NULL)

{

printf("\t\t\t无此学生\n");

system("pause");

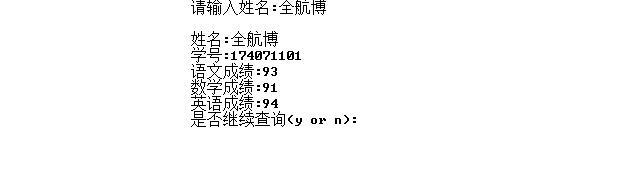
break;

}

}

return 3;

}



int IndexOnNumber(stud \*p)//按学号查询函数

{

if(p==NULL || p->next==NULL)//防止无学生信息

{

printf("\t\t\t请先输入学生信息\n");

system("pause");

return 0;

}

stud \*q=p->next,\*a=p->next;

char n[20],b[20];

char m = 'y';

while(m=='y')

{

printf("\t\t\t请输入学号:");

scanf("%s",b);

printf("\n");

while(q)

{

if(strcmp(q->num,b)==0)//比较节点中的学号是否一致

{ //一致输出信息

printf("\t\t\t姓名:%s\n",q->name);

printf("\t\t\t学号:%s\n",q->num);

printf("\t\t\t语文成绩:%d\n",q->chinese);

printf("\t\t\t数学成绩:%d\n",q->math);

printf("\t\t\t英语成绩:%d\n",q->English);

break;

}

q=q->next;// 不一致指向下一个节点

}

if(q)//如果q！=NULL

{

printf("\t\t\t是否继续查询(y or n):");

scanf(" %c",&m);

}

if (q==NULL)

{

printf("\t\t\t无此学生\n");

system("pause");

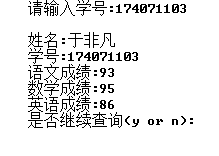
break;

}

}

return 1;

}



（4）计算求总分和平均值思路：

void Computer(stud \*p)//输出学生平均分和总分函数；

返回主界面

进行循环时对（p->sum）/3.0，则为当前学生的平均成绩并输出

在循环里执行p->sum=p->Chinese+p->math+p->english（即总分）并输出

从头指针所指向的数据向后累加直至p->next=NULL

{

if(p==NULL || p->next==NULL)

{

printf("\t\t\t请先输入学生信息\n");

system("pause");

return;

}

stud \*q=p->next;

int totalnumber=0;

while(q)

{

totalnumber++;

printf("\t\t\t第%d个同学总分为：%d\n",totalnumber,q->sum);

printf("\t\t\t 均分为:%.2f\n",(q->sum)/3.0);

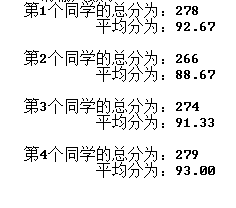
putchar(10);

q=q->next;

}

system("pause");

}



（5）学生成绩排序思路

返回主界面

分情况case（总分，语文，数学，英语）用选择法进行由高到低排序

将结点的数据赋给结构体数组（l[i++]=\*q，\*p=q->next）

结点不为空q!=NULL

stud\* reorder(stud \*p,int x)//排序函数

{ //将链接换成结构体数组，再排序。

stud temp,\*q=p->next;

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

switch(x)

{

case 1: //总分排序

while(q)

{

n++;//计算总数

l[i++]=\*q;//顺序将节点赋给结构体数组元素

q=q->next;

}

for(i=0;i<n-1;i++)

{

k=i;

for(j=i+1;j<n;j++)

{

if(l[k].sum<l[j].sum)

k=j;

if(i!=k)

{

temp=l[i];

l[i]=l[k];

l[k]=temp;

}

}

}

break;

case 2: //语文排序

while(q)

{

n++;

l[i++]=\*q;

q=q->next;

}

for(i=0;i<n-1;i++)

{

k=i;

for(j=i+1;j<n;j++)

{

if(l[k].chinese<l[j].chinese)

k=j;

if(i!=k)

{

temp=l[i];

l[i]=l[k];

l[k]=temp;

}

}

}

break;

case 3: //数学排序

while(q)

{

n++;

l[i++]=\*q;

q=q->next;

}

for(i=0;i<n-1;i++)

{

k=i;

for(j=i+1;j<n;j++)

{

if(l[k].math<l[j].math)

k=j;

if(i!=k)

{

temp=l[i];

l[i]=l[k];

l[k]=temp;

}

}

}

break;

case 4://英语排序

while(q)

{

n++;

l[i++]=\*q;

q=q->next;

}

for(i=0;i<n-1;i++)

{

k=i;

for(j=i+1;j<n;j++)

{

if(l[k].English<l[j].English)

k=j;

if(i!=k)

{

temp=l[i];

l[i]=l[k];

l[k]=temp;

}

}

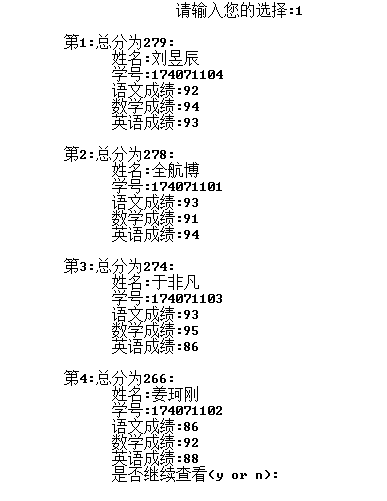
}

break;

}

return (l);

}



(6) 输出学生信息思路：

返回主界面

判断y or n 是否继续输出

利用结构体成员运算符输出成绩

搭建循环i次

int ShowStudent(stud \*p) //输出所有学生信息

{

if(p==NULL || p->next==NULL)//防止无学生信息

{

printf("\t\t\t请先输入学生信息\n");

system("pause");

return 0;

}

stud \*q=p->next;

while(q!=NULL)//当q！=NULL时输出学生信息

{

printf("\t\t\t姓名:%s\n",q->name);

printf("\t\t\t学号:%s\n",q->num);

printf("\t\t\t语文成绩:%d\n",q->chinese);

printf("\t\t\t数学成绩:%d\n",q->math);

printf("\t\t\t英语成绩:%d\n",q->English);

printf("\t\t\t总成绩%d\n\n",q->sum);

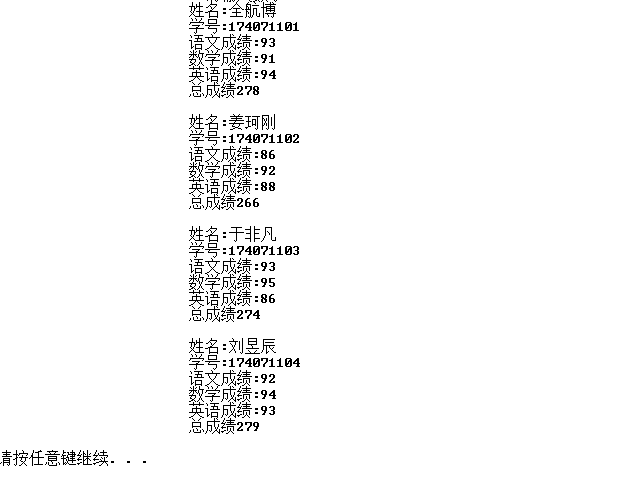
q=q->next;

}

system("pause");

return 1;

}



**3.调试报告**（调试过程中遇到的主要问题，是如何解决的；对设计和编码的回顾讨论和分析；改进设想；经验和体会等）

\*\* 调试过程中的问题如下，以及如何解决的：

(1).如何让系统循环运行，一开始的设计过程中如何将子程序或者主程序循环的运行，由于之前没有设计过这样的系统，没有对应的函数输入不能解决问题。后来在通过同学和网上资料的查找，通过加入while (1) 和break这一组对应的函数解决问题。

(2).刚开始没有想到如何预防用户输入不当的情况，比如学生人数输入为负数，当所指空为空时，开辟一段空间失败等操作。后来经过小组讨论，添加了如果输入有误，则弹出提示信息，指导继续操作。

(3).调试程序时，比如说删除学生信息时，一开始只能删除一个数据不知道用什么方法来连续删除，后来进过上网看别人的程序，用y or n来判断你是否继续删除信息，并把这个程序装在大循环里，这样就实现了删除多个学生数据。

\*\* 改进设想：

(1).在程序中仍然有重复代码，比如说：多次用到输出学生信息的代码（用了printf语句来实现），所以可能的话，可以再写一个函数用来调用，以此来优化代码。

\*\* 经验分享和体会：

(1).通过本次C语言课程设计，进一步理解了如何使用结构体，链表，函数，的综合应用，更加体会到了C语言和实践生活的联系，又一次极大地提升了对编程的兴趣。

(2).更加深刻的认识到，写代码和调试代码真的需要极大的耐心，不懂就要问，哪怕是很小的失误都会导致程序出错或者出现无法预计的后果。

(3).通过写代码，才意识到自己真的很差劲，然后小组讨论以及上网查代码，不仅让我认识到了一个优秀的团队的重要性以及自己不会的东西，通过阅读别人的优秀代码，来不断提升自己。

(4).虽然编程是一件很累的事，但是为了更好掌握优秀的知识，我们更需要不断改代码，优化代码，使自己的编程能力不断提升。

(5).最后非常感谢几位老师的指导，谢谢！

4.**附录** (源代码)

#include<stdio.h>

#include<stdlib.h>

#include<windows.h>

#include<string.h>

typedef struct student {

char num[20];//学号

char name[20];

int chinese;

int math;

int English;

int sum;//总分

struct student \*next;

} stud;

stud l[50];

int menu\_select();//菜单选择函数

int circulate();//循环选择函数

stud\* Creat();//录入模块

int ShowStudent(stud \*p);//显示模块

int Search(stud \*p);//查询模块

int IndexOnNumber(stud \*p);//查询模块

stud\* reorder(stud \*p,int x);//排序函数

char print(stud \*d,int n);//输出函数

void Computer(stud \*p);//统计模块

void End();//结束函数

int main()

{

system("color F0");//改颜色

circulate();

return 0;

}

int menu\_select() {//选择函数

int c;

system("cls");

printf("\n");

printf("\t\t 欢迎使用学生成绩管理系统 \n");

printf("\t\t \n");

printf("\t\t 1.输入学生信息 \n");

printf("\t\t \n");

printf("\t\t 2.删除学生信息 \n");

printf("\t\t \n");

printf("\t\t 3.增添学生信息 \n");

printf("\t\t \n");

printf("\t\t 4.显示所有学生信息 \n");

printf("\t\t \n");

printf("\t\t 5.按查姓名查找学生信息 \n");

printf("\t\t \n");

printf("\t\t 6.计算总分和平均分 \n");

printf("\t\t \n");

printf("\t\t 7.学生成绩排名 \n");

printf("\t\t \n");

printf("\t\t 8.按查学号查找学生信息 \n");

printf("\t\t \n");

printf("\t\t 9.退出系统 \n");

printf("\n");

printf("\t\t (请输入数字1~9): ");

scanf("%d",&c);

return(c);

}

stud\* Creat() //创建函数

{

system("cls");

printf("\t\t HELLO，您正在输入学生信息... \n\n");

printf("\t\t 请根据提示输入学生信息: \n\n");

int i,j;

stud \*head,\*p1,\*p2;

printf("\t\t\t ◎请输入总数:");

scanf("%d",&i);

if(i<=0)

{

printf("\t\t\t输入错误请重试\n");

system("pause");//暂停屏幕

}

if((head=(stud \*)malloc(sizeof(stud)))==NULL)//创建头指针

{

printf("\t\t\t空间不足建立失败\n");

exit(0);

}

head->next=NULL;

p1=head;

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

{

if((p2=(stud \*)malloc(sizeof(stud)))==NULL)//创建新的节点

{

printf("\t\t\t空间不足建立失败\n");

exit(0);

}

//输入学生信息

printf("\t\t您正在输入第%d个人的信息\n",j+1);

printf("\t\t ◎请输入学号:");

scanf("%s",p2->num);

printf("\t\t ◎请输入姓名:");

scanf("%s",p2->name);

printf("\t\t ◎请输入语文成绩:");

scanf("%d",&p2->chinese);

printf("\t\t ◎请输入数学成绩:");

scanf("%d",&p2->math);

printf("\t\t ◎请输入英语成绩:");

scanf("%d",&p2->English);

p2->sum=p2->chinese+p2->English+p2->math;

//使上一个节点指向下一个节点

p1->next=p2;

p2->next=NULL;//使最后节点next为空

p1=p2;

if(i>=3 && (j+1)%2==0) system("cls");

}

printf("\t\t已成功输入学生信息\n");

system("pause");

return (head);//返回头指针

}

int Search(stud \*p) //按姓名查询函数

{ //传头指针过来

if(p==NULL || p->next==NULL)//防止无学生信息

{

printf("\t\t\t请先输入学生信息\n");

system("pause");

return 0;

}

stud \*q=p->next,\*a=p->next;

char n[20],b[20];

char m = 'y';

while(m == 'y')//当m=‘y’时循环

{

q=a;//每次循环重置指针q

system("cls");

printf("\t\t\t请输入姓名:");

scanf("%s",b);

printf("\n");

while(q)

{

if(strcmp(q->name,b)==0)//比较节点中的姓名是否一致

{

printf("\t\t\t姓名:%s\n",q->name);

printf("\t\t\t学号:%s\n",q->num);

printf("\t\t\t语文成绩:%d\n",q->chinese);

printf("\t\t\t数学成绩:%d\n",q->math);

printf("\t\t\t英语成绩:%d\n",q->English);

break;

}

else q=q->next;

}

if(q)//等同于q!=NULL;

{

printf("\t\t\t是否继续查询(y or n):");

scanf(" %c",&m);

}

if (q==NULL)

{

printf("\t\t\t无此学生\n");

system("pause");

break;

}

}

return 3;

}

int IndexOnNumber(stud \*p)//按学号查询函数

{

if(p==NULL || p->next==NULL)//防止无学生信息

{

printf("\t\t\t请先输入学生信息\n");

system("pause");

return 0;

}

stud \*q=p->next,\*a=p->next;

char n[20],b[20];

char m = 'y';

while(m=='y')

{

printf("\t\t\t请输入学号:");

scanf("%s",b);

printf("\n");

while(q)

{

if(strcmp(q->num,b)==0)//比较节点中的学号是否一致

{ //一致输出信息

printf("\t\t\t姓名:%s\n",q->name);

printf("\t\t\t学号:%s\n",q->num);

printf("\t\t\t语文成绩:%d\n",q->chinese);

printf("\t\t\t数学成绩:%d\n",q->math);

printf("\t\t\t英语成绩:%d\n",q->English);

break;

}

q=q->next;// 不一致指向下一个节点

}

if(q)//如果q！=NULL

{

printf("\t\t\t是否继续查询(y or n):");

scanf(" %c",&m);

}

if (q==NULL)

{

printf("\t\t\t无此学生\n");

system("pause");

break;

}

}

return 1;

}

int ShowStudent(stud \*p) //输出所有学生信息

{

if(p==NULL || p->next==NULL)//防止无学生信息

{

printf("\t\t\t请先输入学生信息\n");

system("pause");

return 0;

}

stud \*q=p->next;

while(q!=NULL)//当q！=NULL时输出学生信息

{

printf("\t\t\t姓名:%s\n",q->name);

printf("\t\t\t学号:%s\n",q->num);

printf("\t\t\t语文成绩:%d\n",q->chinese);

printf("\t\t\t数学成绩:%d\n",q->math);

printf("\t\t\t英语成绩:%d\n",q->English);

printf("\t\t\t总成绩%d\n\n",q->sum);

q=q->next;

}

system("pause");

return 1;

}

int Add(stud \*p)//添加函数

{

if(p==NULL || p->next==NULL)

{

printf("\t\t\t请先输入学生信息\n");

system("pause");

return 4;

}

stud \*q=p,\*a=p,\*head[100];

int i=-1,b,n=0;

char m='y';

while(q->next)

{

n++;

q=q->next;

}

while(m=='y')

{

system("cls");

q=a;

printf("\t\t\t您正在添加学生信息...:\n");

if((head[++i]=(stud \*)malloc(sizeof(stud)))==NULL)

{

printf("\t\t\t空间不足建立失败\n");

exit(0);

}

printf("\t\t\t需要插入到第b个数据之后(0<=b<=%d):",n);

scanf("%d",&b);

if(b<0 || b>n)

{

printf("\n\t\t输入错误\n");

system("pause");

break;

}

//插入到第b个数据之后

for(n=1;n<=b;n++)

q=q->next;

head[i]->next=NULL;

printf("\t\t◎请输入学号:");

scanf("%s",head[i]->num);

printf("\t\t◎请输入姓名:");

scanf("%s",head[i]->name);

printf("\t\t◎请输入语文成绩:");

scanf("%d",&head[i]->chinese);

printf("\t\t◎请输入数学成绩:");

scanf("%d",&head[i]->math);

printf("\t\t◎请输入英语成绩:");

scanf("%d",&head[i]->English);

head[i]->sum=head[i]->chinese+head[i]->English+head[i]->math;

head[i]->next=q->next;

q->next=head[i];

printf("\t\t\t输入成功\n");

printf("\t\t\t是否继续增添(y or n):");

scanf(" %c",&m);

}

return 1;

}

int Del(stud \*p) //删除函数

{

if(p==NULL || p->next==NULL)

{

printf("\t\t\t请先输入学生信息\n");

system("pause");

return 0;

}

stud \*q=p->next,\*a=p->next,\*h=p;//h为q的前一个节点

char m='y',b[20],n[20];

int j;

while(m=='y')

{

if(p==NULL)

{

printf("\t\t\t请先输入学生信息\n");

system("pause");

return 0;

}

system("cls");

q=a;

h=p;

printf("\t\t 您正在删除学生信息 ... \n");

printf("\t\t \n");

printf("\t\t 请选择删除方式 \n");

printf("\t\t \n");

printf("\t\t 1.按学号删除 \n");

printf("\t\t \n");

printf("\t\t 2.按姓名删除 \n");

printf("\t\t \n");

printf("\t\t 0.退出 \n");

printf("\n");

printf("\t\t 请输入您的选择:");

scanf("%d",&j);

switch(j)

{

case 1:

printf("\t\t\t请输入学号:");

scanf("%s",b);

while(q)

{

if(strcmp(q->num,b)==0)

{

h->next=q->next;//使前一个节点指向后一个

free(q);

printf("\t\t\t删除成功\n");

system("pause");

break;

}

h=q;//h为q的前一个节点

q=q->next;

}

if (q==NULL)

{

printf("\t\t\t无此学生\n");

system("pause");

break;

}

if(q)

{

printf("\t\t\t是否继续删除(y or n):");

scanf(" %c",&m);

}

break;

case 2:

printf("\t\t\t请输入姓名:");

scanf("%s",n);

while(q)

{

if(strcmp(q->name,n)==0)

{

h->next=q->next;//使前一个节点指向后一个

free(q);

printf("\t\t\t删除成功\n");

break;

}

h=q;

q=q->next;

}

if (q==NULL)

{

printf("\t\t\t无此学生\n");

system("pause");

break;

}

if(q)

{

printf("\t\t\t是否继续删除(y or n):");

scanf(" %c",&m);

}

break;

case 0:

m='n';

break;

default:

printf("\t\t\t输入错误\n");

system("pause");

break;

}

}

return 1;

}

void Computer(stud \*p)//输出学生平均分和总分函数；

{

if(p==NULL || p->next==NULL)

{

printf("\t\t\t请先输入学生信息\n");

system("pause");

return;

}

stud \*q=p->next;

int totalnumber=0;

while(q)

{

totalnumber++;

printf("\t\t\t第%d个同学总分为：%d\n",totalnumber,q->sum);

printf("\t\t\t 均分为:%.2f\n",(q->sum)/3.0);

putchar(10);

q=q->next;

}

system("pause");

}

char print(stud \*d,int n)//输出学生全部信息函数；

{

char m='y';

int i;

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

{

printf("\n\t\t 第%d:",i+1);

printf("总分为%d:\n",d[i].sum);

printf("\t\t\t姓名:%s\n",d[i].name);

printf("\t\t\t学号:%s\n",d[i].num);

printf("\t\t\t语文成绩:%d\n",d[i].chinese);

printf("\t\t\t数学成绩:%d\n",d[i].math);

printf("\t\t\t英语成绩:%d\n",d[i].English);

}

printf("\t\t\t是否继续查看(y or n):");

scanf(" %c",&m);

return m;

}

stud\* reorder(stud \*p,int x)//排序函数

{ //将链接换成结构体数组，再排序。

stud temp,\*q=p->next;

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

switch(x)

{

case 1: //总分排序

while(q)

{

n++;//计算总数

l[i++]=\*q;//顺序将节点赋给结构体数组元素

q=q->next;

}

for(i=0;i<n-1;i++)

{

k=i;

for(j=i+1;j<n;j++)

{

if(l[k].sum<l[j].sum)

k=j;

if(i!=k)

{

temp=l[i];

l[i]=l[k];

l[k]=temp;

}

}

}

break;

case 2: //语文排序

while(q)

{

n++;

l[i++]=\*q;

q=q->next;

}

for(i=0;i<n-1;i++)

{

k=i;

for(j=i+1;j<n;j++)

{

if(l[k].chinese<l[j].chinese)

k=j;

if(i!=k)

{

temp=l[i];

l[i]=l[k];

l[k]=temp;

}

}

}

break;

case 3: //数学排序

while(q)

{

n++;

l[i++]=\*q;

q=q->next;

}

for(i=0;i<n-1;i++)

{

k=i;

for(j=i+1;j<n;j++)

{

if(l[k].math<l[j].math)

k=j;

if(i!=k)

{

temp=l[i];

l[i]=l[k];

l[k]=temp;

}

}

}

break;

case 4://英语排序

while(q)

{

n++;

l[i++]=\*q;

q=q->next;

}

for(i=0;i<n-1;i++)

{

k=i;

for(j=i+1;j<n;j++)

{

if(l[k].English<l[j].English)

k=j;

if(i!=k)

{

temp=l[i];

l[i]=l[k];

l[k]=temp;

}

}

}

break;

}

return (l);

}

int Compare(stud \*p) //比较函数

{

if(p==NULL || p->next==NULL)

{

printf("\t\t\t请先输入学生信息\n");

system("pause");

return 0;

}

stud \*q=p->next,\*a=p->next,\*d=NULL;

char m='y';

int n=0,f,x;

while(q)

{

n++;//计算总数

q=q->next;

}

while(m=='y')

{

system("cls");

printf("\t\t 您正在查看学生排名:\n\n");

printf("\t\t 请选择查看方式\n");

printf("\t\t 1.查询总分排序\n");

printf("\t\t 2.查询语文排序\n");

printf("\t\t 3.查询数学排序\n");

printf("\t\t 4.查询英语排序\n");

printf("\t\t 0.退出\n");

printf("\n");

printf("\t\t 请输入您的选择:");

scanf("%d",&f);

switch(f)

{

case 1:

d=reorder(p,x=1) ;//相应的x的值对应相应的排序

m=print(d,n);

break;

case 2:

d=reorder(p,x=2) ;

m=print(d,n);

break;

case 3:

d=reorder(p,x=3) ;

m=print(d,n);

break;

case 4:

d=reorder(p,x=4) ;

m=print(d,n);

break;

case 0:

m='n';

break;

default:

printf("\t\t\t输入错误\n");

system("pause");

break;

}

}

return 1;

}

void End()

{

system("cls");

printf("感谢使用学生管理系统,再见！\n");

printf("\n");

system("pause");

}

int circulate()

{

stud \*p=NULL;

while(1) {

switch(menu\_select()) {

case 1:

p=Creat();

break;

case 2:

Del(p);

break;

case 3:

Add(p);

break;

case 4:

ShowStudent(p);

break;

case 5:

Search(p);

break;

case 6:

Computer(p);

break;

case 7:

Compare(p);

break;

case 8:

IndexOnNumber(p);

break;

case 9:

End();

return 0;

default :

printf("\t\t\t输入错误请重新输入\n");

printf("\n");

system("pause");

break;

}

}