#include<stdio.h>

#include<stdlib.h>

#include<string.h>

struct student

{

int num,pos;

double grade;

char name[20],sex[10],depart[20],type[20];

char unit ;

struct student \*next;

};

struct student search[100];

int p=0; //定义个人信息数据creat()

struct student \* creat()

{

struct student \* head,\*tail,\*nnode;

FILE \* fp;

fp =fopen ("C:\\Users\\lianxiang\\Desktop\\result1.txt","r");

/\*printf ("无法打开数据文件，没有本地数据！\n");

return NULL;

}\*/

if ((head = (struct student\*)malloc (sizeof (struct student)))==NULL)

printf ("无法创建链表！\n");

head ->next=NULL;

tail=head;

char name [20],sex [10],depart [20],type [20],unit;

int n = 0,m;

double grade;

while ((fscanf(fp,"%d %s %s %s %s %lf %c",&m,name,sex,depart,type,&grade,&unit)!=EOF)){

++n;

if ((nnode = (struct student \*)malloc(sizeof(struct student))) == NULL)

printf("无法创建链表！\n");

nnode->next=NULL;

strcpy(nnode->name, name);

strcpy(nnode->sex, sex);

strcpy(nnode->depart, depart);

strcpy(nnode->type, type);

nnode->grade = grade;

nnode->unit = unit;

nnode->num = n;

nnode->pos=m;

tail->next=nnode;

tail=nnode;

}

head->num=n;

fclose (fp);

struct student \* creat()

{

struct student \* head,\*tail,\*nnode;

FILE \* fp;

fp =fopen ("C:\\Users\\lianxiang\\Desktop\\result1.txt","r");

/\*printf ("无法打开数据文件，没有本地数据！\n");

return NULL;

}\*/

if ((head = (struct student\*)malloc (sizeof (struct student)))==NULL)

printf ("无法创建链表！\n");

head ->next=NULL;

tail=head;

char name [20],sex [10],depart [20],type [20],unit;

int n = 0,m;

double grade;

while ((fscanf(fp,"%d %s %s %s %s %lf %c",&m,name,sex,depart,type,&grade,&unit)!=EOF)){

++n;

if ((nnode = (struct student \*)malloc(sizeof(struct student))) == NULL)

printf("无法创建链表！\n");

nnode->next=NULL;

strcpy(nnode->name, name);

strcpy(nnode->sex, sex);

strcpy(nnode->depart, depart);

strcpy(nnode->type, type);

nnode->grade = grade;

nnode->unit = unit;

nnode->num = n;

nnode->pos=m;

tail->next=nnode;

tail=nnode;

}

head->num=n;

fclose (fp);

return (head);

}

return (head);

}

struct student \*del( struct student \* head,int pos)

{

struct student \*p1,\*p2;

if (head==NULL)

printf ("list null!\n");

p1=head;

while (pos!=p1->pos&&p1->next!=NULL){

p2=p1;

p1=p1->next;

}

if (pos==p1->pos){

if (p1==head) head=p1->next;

else p2->next=p1->next;

head->num-=1;

}

else printf ("找不到该结点!\n");

return (head);

}

void print (struct student \*head)

{

struct student \*p;

int i=0;

for (p = head->next; (p != NULL) &&( i < head->num); p = p->next, ++i)

printf("%d %s %s %s %s %lf %c\n", p->pos,p->name, p->sex, p->depart, p->type, p->grade,p->unit);

}

void research (struct student \*head,char name[])

{ struct student \*p1,\*p2;

FILE \* in;

p=0;

if (head==NULL)

printf ("list null!\n");

p1=head;

for (p1=head;p1!=NULL;p1=p1->next){

if (!strcmp(name,p1->name)){

strcpy(search[p].name, p1->name);

strcpy(search[p].sex, p1->sex);

strcpy(search[p].depart,p1->depart);

strcpy(search[p].type,p1->type);

search[p].grade = p1->grade;

search[p].unit =p1-> unit;

search[p].pos= p1->pos;

search[p].num=p1->num;

p++;

}

}

}

modify (struct student \* head,int num,double grade)

{ int i;

struct student \*p1;

for (p1=head;p1!=NULL;p1=p1->next){

if (p1->==num){

p1->grade=grade;

break;

}

}

}

int main()

{

struct student\*head=NULL;

FILE \* ll;

int m;

double grade;

char lname[20],sex[10],depart[20],type[20];

char unit ;

head = creat ();

int del\_num,i;

char name [20];

printf ("请输入要删除的编号：\n");

scanf ("%d",&del\_num);

while (del\_num!=0){

head=del(head,del\_num);

print(head);

printf ("input the deleted number:\n");

scanf ("%d",&del\_num);

}

printf ("输入要查询的姓名:\n");

scanf ("%s",name);

research (head,name);

for (i=0;i<p;i++){

printf("%d %s %s %s %s %lf %c\n", search[i].pos,search[i].name,

search[i].sex, search[i].depart, search[i].type, search[i].grade,search[i].unit);

}

int k;

double lgrade;

char llname[20],lsex[10],ldepart[20],ltype[20];

char lunit ;

FILE \* LL;

LL=fopen("C:\\Users\\lianxiang\\Desktop\\result1.txt","a");

printf ("请输入要增加的人员信息：\n");

scanf("%d %s %s %s %s %lf %c",&k,llname,lsex,ldepart,ltype,&lgrade,&lunit);

fprintf (LL,"%d %s %s %s %s %lf %c\n",k,llname,lsex,ldepart,ltype,lgrade,lunit);

fclose(LL);