实验二 通讯录

#include <iostream>

#include <string.h>

using namespace std;

typedef struct

{

char num[30];

char name[30];

char city[30];

char phone[30];

}DataType;

typedef struct Node

{

DataType data;

struct Node \*next;

}ListNode, \*LinkList;

void IintList\_L(LinkList &L)

{

L=NULL;

cout<<"初始化成功"<<endl;

system("pause");

}

void CreateList\_L(LinkList &L)

{

int i,n;

ListNode \*p;

L=(LinkList) malloc (sizeof (Node));

L->next=NULL;

cout<<"请输入学生人数:"<<endl;

cin>>n;

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

{

p=(LinkList) malloc (sizeof (Node));

cout<<"输入第"<<i+1<<"位学生姓名:"<<endl;

cin>>p->data.name;

cout<<"输入第"<<i+1<<"位学生城市:"<<endl;

cin>>p->data.city;

cout<<"输入第"<<i+1<<"位学生电话号码:"<<endl;

cin>>p->data.phone;

cout<<"输入第"<<i+1<<"位学生学号:"<<endl;

cin>>p->data.num;

p->next=L->next;

L->next=p;

}

system("pause");

}

void SearchnamedataList\_L(LinkList &L)

{

ListNode \*p;

char name[30];

int j=0;

cout<<"请输入要查找的姓名:"<<endl;

cin>>name;

p=L->next;

while(p)

{

if(strcmp(p->data.name,name)==0)

{

j=1;

break;

}

p=p->next;

}

if(j)

{

cout<<"姓名"<<p->data.name<<endl;

cout<<"城市"<<p->data.city<<endl;

cout<<"电话号码"<<p->data.phone<<endl;

cout<<"学号"<<p->data.num<<endl;

}

else

cout<<"姓名为"<<name<<"的学生不存在"<<endl;

system("pause");

}

void SearchnumdataList\_L(LinkList &L)

{

ListNode \*p;

char num[30];

int j=0;

cout<<"请输入要查找的学号:"<<endl;

cin>>num;

p=L->next;

while(p)

{

if(strcmp(p->data.num,num)==0)

{

j=1;

break;

}

p=p->next;

}

if(j)

{

cout<<"姓名"<<p->data.name<<endl;

cout<<"城市"<<p->data.city<<endl;

cout<<"电话号码"<<p->data.phone<<endl;

cout<<"学号"<<p->data.num<<endl;

}

else

cout<<"学号为"<<num<<"的学生不存在"<<endl;

system("pause");

}

void ModifydataList\_L(LinkList &L)

{

ListNode \*p;

int j=0;

p=L->next;

char num[30];

cout<<"请输入需要修改信息学生的学号:"<<endl;

cin>>num;

while(p)

{

if(strcmp(p->data.num,num)==0)

{

j=1;

break;

}

p=p->next;

}

if(j)

{

cout<<"输入修改后学生的姓名:"<<endl;

cin>>p->data.name;

cout<<"输入修改后学生的城市:"<<endl;

cin>>p->data.city;

cout<<"输入修改后学生的电话号码:"<<endl;

cin>>p->data.phone;

}

else

cout<<"学号为"<<num<<"的学生不存在"<<endl;

system("pause");

}

void insertdataList\_L(LinkList &L)

{

ListNode \*p,\*q;

p=L;

q=(LinkList) malloc (sizeof (Node));

cout<<"输入需要插入学生的学号:"<<endl;

cin>>q->data.num;

cout<<"输入需要插入学生的姓名:"<<endl;

cin>>q->data.name;

cout<<"输入需要插入学生的城市:"<<endl;

cin>>q->data.city;

cout<<"输入需要插入学生的电话号码:"<<endl;

cin>>q->data.phone;

while(p->next!=NULL)

p=p->next;

q->next=p->next;

p->next=q;

system("pause");

}

void DeletedataList\_L(LinkList &L)

{

ListNode \*p,\*q;

char num[30];

int j=0;

p=L;

cout<<"输入需要删除学生的学号:"<<endl;

cin>>num;

while(p->next)

{

if(strcmp(p->next->data.num,num)==0)

{

j=1;

break;

}

p=p->next;

}

if(j)

{

q=p->next;

p->next=q->next;

free(q);

cout<<"删除成功"<<endl;

}

else

cout<<"学号为"<<num<<"的学生不存在"<<endl;

system("pause");

}

void TraverserList\_L(LinkList &L)

{

ListNode \*p;

if(L==NULL)

{

cout<<"通讯录为空"<<endl;

return;

}

else

{

p=L->next;

cout<<"通讯录的信息如下:"<<endl;

while(p)

{

cout<<"学号:"<<p->data.num<<endl;

cout<<"姓名:"<<p->data.name<<endl;

cout<<"城市:"<<p->data.city<<endl;

cout<<"电话号码:"<<p->data.phone<<endl;

p=p->next;

}

}

system("pause");

}

void lengthList\_L(LinkList &L)

{

int n=0;

ListNode \*p;

p=L;

while(p->next!=NULL)

{

n++;

p=p->next;

}

cout<<"通讯录人数为:"<<n<<endl;

system("pause");

}

void main()

{

int g;

LinkList L;

cout<<"-----欢迎使用通讯录-----"<<endl;

cout<<"创建通讯录:"<<endl;

CreateList\_L(L);

while(true)

{

cout<<"-------1.以学号查询----"<<endl;

cout<<"-------2.以姓名查询----"<<endl;

cout<<"-------3.修改----------"<<endl;

cout<<"-------4.删除----------"<<endl;

cout<<"-------5.插入----------"<<endl;

cout<<"-------6.遍历通讯录----"<<endl;

cout<<"-------7.初始化通讯录--"<<endl;

cout<<"-------8.通讯录人数----"<<endl;

cout<<"-------9.退出通讯录----"<<endl;

cout<<"请选择功能："<<endl;

cin>>g;

switch(g)

{

case 1:

{

SearchnumdataList\_L(L);

break;

}

case 2:

{

SearchnamedataList\_L(L);

break;

}

case 3:

{

ModifydataList\_L(L);

break;

}

case 4:

{

DeletedataList\_L(L);

break;

}

case 5:

{

insertdataList\_L(L);

break;

}

case 6:

{

TraverserList\_L(L);

break;

}

case 7:

{

IintList\_L(L);

break;

}

case 8:

{

lengthList\_L(L);

break;

}

}

if(g==9)

{

cout<<"退出通讯录"<<endl;

break;

}

}

}