第十一讲单元作业

1、编写函数deln，具有删除链表中第n个结点的功能。再编写主函数，按输入顺序建立不带头结点的职工信息单链表，然后调用del函数删除某个职工的信息，并输出删除后的职工信息表中的职工信息。假设链表结构如下：

struct staff

{

char num[6]; //职工工号

char name[20]; //职工姓名

double wage; //职工工资

};

答案1：

#define N 5

#include <iostream>

using namespace std;

struct node

{

char num[6]; //职工工号

char name[20]; //职工姓名

double wage; //职工工资

node \*next;

};

node \*deln(node \*head,int n)

{

node \*p=head,\*q;

int i=1;

if(n==1)

head=head->next;

else

{ while(i<n&&p->next!=NULL)

{

q=p;

p=p->next;

i++;

}

q->next =p->next ;

}

delete p;

return head;

}

void print(node \*head)

{

node \*p=head;

while(p!=NULL)

{

cout<<p->name<<' '<<p->num <<' '<<p->wage <<endl;

p=p->next;

}

}

int main()

{

node \*head=NULL,\*tail=NULL,\*newnode;

int i,n;

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

{

newnode=new node;

cin>>newnode->name >>newnode->num>>newnode->wage ;

if(head==NULL)

head=newnode;

else

tail->next=newnode;

tail=newnode;

}

tail->next=NULL;

print(head);

cin>>n; //输入待删除结点的序号

head=deln(head,n);

print(head);

system("pause");

return 0;

}

2. 在上一题目建立的链表中增加一个按工号num修改该职工工资wage的函数。再编写主函数，对某一具体职工的信息进行更改。

答案2：

#define N 5

#include <iostream>

using namespace std;

struct node

{

char num[6]; //职工工号

char name[20]; //职工姓名

double wage; //职工工资

node \*next;

};

void modify(node \*head,char num[],double w)

{

node \*p=head;

while(p!=NULL&&strcmp(p->num ,num)!=0)

p=p->next ;

if(p==NULL)

cout<<num<<"not exist.\n";

else

p->wage =w;

}

void print(node \*head)

{

node \*p=head;

while(p!=NULL)

{

cout<<p->name<<' '<<p->num <<' '<<p->wage <<endl;

p=p->next;

}

}

int main()

{

node \*head=NULL,\*tail=NULL,\*newnode;

int i;

char num[6];

double w;

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

{

newnode=new node;

cin>>newnode->name >>newnode->num>>newnode->wage ;

if(head==NULL)

head=newnode;

else

tail->next=newnode;

tail=newnode;

}

tail->next=NULL;

print(head);

cin>>num>>w; //输入待修改信息的工号、新工资

modify(head,num,w);

print(head);

system("pause");

return 0;

}

3.编写程序建立一个单向链表。链表结点中的数据为从键盘输入的一个字符串，但要求将该串字符按由小到大的顺序组织在链表中。

答案3：

#include <iostream>

using namespace std;

struct node

{

char ch;

node \*next;

};

int main()

{

struct node \*head, \*p, \*q,\*newnode,\*tail;

char s[100];

gets(s);

head = new node ; // 在链表中插入伪结点，不关心该结点的数据域的值。

head->next =NULL;

tail = head;

for (int i=0;s[i]!='\0';i++)

{

newnode=new node;

newnode->ch=s[i];

newnode->next =NULL;

p=head->next;

while(p!=NULL&&s[i]>p->ch)

{

q=p;

p=p->next ;

}

if(p==NULL)

{

tail->next =newnode;

tail=newnode;

}

else

{

q->next =newnode;

newnode->next =p;

}

}

p=head->next ;

while(p!=NULL)

{

cout<<p->ch;

p=p->next ;

}

system("pause");

return 0;

}