#include<iostream>

using namespace std;

class Node

{

private:

int data;

Node \*next;

public:

Node()

{

}

Node(int data)

{

this->data=data;

this->next=NULL;

}

int getData()

{

return data;

}

Node \*getNext()

{

return next;

}

void setData(int data)

{

}

void setNext(Node \*next)

{

this->next=next;

}

void print()

{

if(data!=0)

cout<<data<<"->";

}

};

class Linklist

{

Node \*head;

public:

Linklist()

{

head=NULL;

}

void insertAtEnd(int data)

{

//cout<<"data="<<data<<endl;

Node \*newnode=new Node(data);

Node \*temp=new Node();

temp=head;

if(head==NULL)

{

head=newnode;

}

else

{

while(temp->getNext()!=NULL)

{

temp=temp->getNext();

}

temp->setNext(newnode);

}

}

void insertBefore(int before,int data)

{

Node \*newnode=new Node(data);

Node \*temp=head;

Node \*prev=NULL;

while(temp!=NULL)

{

if(temp->getData()==before)

{

break;

}

prev=temp;

temp=temp->getNext();

}

if(prev==NULL)

{

insertAtBeg(data);

}

else

{

prev->setNext(newnode);

newnode->setNext(temp);

}

}

void insertAfter (int after,int data)

{

//cout<<"inside insertAfter"<<endl;

Node \*newnode=new Node(data);

Node \*temp=head;

while(temp!=NULL)

{

//cout<<"inside while"<<endl;

if(temp->getData()==after)

{

break;

}

temp=temp->getNext();

}

if(temp->getNext()==NULL)

{

insertAtEnd(data);

}

else

{

newnode->setNext(temp->getNext());

temp->setNext(newnode);

}

}

void insertAtBeg(int data)

{

Node \*newnode=new Node(data);

Node \*temp=head;

if(head==NULL)

{

head=newnode;

}

else

{

newnode->setNext(head);

head=newnode;

}

}

void deleteAtEnd()

{

if(head==NULL){

cout<<"list is empty";

}

if(head->getNext()==NULL){

delete head;

//return NULL;

}

Node \*temp=head;

Node \*prev=head;

while(temp->getNext()!=NULL){

prev=temp;

temp=temp->getNext();

}

prev->setNext(NULL);

}

void deleteAtBeg()

{

if(head==NULL){

cout<<"list is empty";

}

if(head->getNext()==NULL){

delete head;

}

Node \*temp=head;

head=temp->getNext();

}

void deleteAfter(int after)

{

//cout<<"inside delete after"<<endl;

Node \*temp=head;

Node \*prev=NULL;

if(head==NULL)

cout<<"list is empty";

if(head->getNext()==NULL)

deleteAtEnd();

while(temp!=NULL){

prev=temp;

if(temp->getData()==after){

break;

}

temp=temp->getNext();

}

temp=temp->getNext();

prev->setNext(temp->getNext());

}

void deleteBefore(int before)

{

if(head==NULL)

cout<<"list is empty";

if(head->getNext()==NULL)

deleteAtBeg();

Node \*temp=head;

Node \*prev=NULL;

while(temp!=NULL)

{

if(temp->getNext()!=NULL){

if(temp->getNext()->getData()==before){

if(temp==head)

{

prev=temp;

temp=temp->getNext();

head=temp;

delete prev;

}

else

break;

}

}

prev=temp;

temp=temp->getNext();

}

if(prev!=NULL && temp!=NULL)

prev->setNext(temp->getNext());

}

void print()

{

Node \*temp=new Node();

temp=head;

while(temp!=NULL)

{

temp->print();

temp=temp->getNext();

}

cout<<"Null"<<endl;

}

};

void list()

{

cout<<"a-insert at end"<<endl;

cout<<"b-insert at begining"<<endl;

cout<<"p-print list"<<endl;

cout<<"c-insert before"<<endl;

cout<<"d-insert after"<<endl;

cout<<"e-delete at end"<<endl;

cout<<"f-delete at begining"<<endl;

cout<<"g-delete after";

cout<<"x-exit"<<endl;

cout<<"l-list"<<endl;

}

int main()

{

Linklist ll;

while(1)

{

char ch;

cin>>ch;

if(ch=='l')

list();

if(ch=='x')

break;

if(ch=='a')

{

int i;

cin>>i;

ll.insertAtEnd(i);

}

if(ch=='b')

{

int i;

cin>>i;

ll.insertAtBeg(i);

}

if(ch=='p')

{

ll.print();

}

if(ch=='c')

{

int b,d;

cin>>b>>d;

ll.insertBefore(b,d);

}

if(ch=='d')

{

int a,d;

cin>>a>>d;

ll.insertAfter(a,d);

}

if(ch=='e')

{

ll.deleteAtEnd();

}

if(ch=='f')

{

ll.deleteAtBeg();

}

if(ch=='g')

{

int a;

cin>>a;

ll.deleteAfter(a);

}

if(ch=='h')

{

int b;

cin>>b;

ll.deleteBefore(b);

}

}

return 0;

}