#include<stdio.h>

#define N 100

typedef int ElemType;//声明ElemType为整型

typedef struct list{

ElemType data[N];//定义 ElemType为整型数组有100个元素

int len;//数组中实际元素的个数

}list;

//初始化

void init(list\* l)//定义inint函数

{

int n,i;

l->len=0;

printf("数组最大长度为%d，请输入元素个数：\n",N);

scanf("%d",&n);

printf("请输入数组元素\n");

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

{scanf("%d",&l->data[i]);

l->len++;

}

}

void addValue(list\* l,int index,ElemType e)//定义addValue函数

//index表示插入位置，插入元素e

{

if(index>l->len){

printf("插入位置大于数组元素个数，插入失败\n");

return;

}

for (int i =l->len-1 ; i >=index ; i--) {

l->data[i+1]= l->data[i];

}

l->data[index]= e;

l->len++;

}

int deleteValue(list \*l, ElemType value) { //定义deleteValue函数

int count=0;

for (int j=0; j<l->len-count;)

{ if (l->data[j] == value)

{

for (int i = j;i<l->len-1;i++)

{

l->data[i] = l->data[i+1];

}

count++;l->len--;

}

else {

j++;

}

}

return count;

}

bool isExitValue(list l, int value)

{

for (int i = 0; i < l.len; i++)

{

if (l.data[i] == value)

{

return true;

}

}

return false;

}

void printList(list l)

{int i;

for (int i = 0; i < l.len; i++)

{

printf("%d\n", l.data[i]);

}

}

int main() {

list p;

init(&p);

printf("初始化后数组元素为\n");

printList(p);

int m;int n;

printf("请输入要插入数据的位置和元素值\n");

scanf("%d%d",&m,&n);

printf("插入数据位置:%d,元素值：%d\n",m,n);

addValue(&p,m,n);

printList(p);

printf("请输入要查找的数据\n");

scanf("%d",&m);

if(isExitValue(p,m))printf("数据存在\n");

else printf("数据不存在\n");

printf("请输入要删除的数据\n");

scanf("%d",&m);

printf("删除了%d个\n",deleteValue(&p,m));

printf("删除后数组元素的值是\n");

printList(p);

return 0;

}