**Danh sách số nguyên cài đặt bằng mảng**

#include<stdio.h>

#include<conio.h>

#include<stdlib.h>

#define N 100 //số phần tử tối đa là 100

**// bieu dien danh sach bang mang**

typedef int item;

typedef struct{

item a[N]; //mang kieu int

int count; //biến đếm các phần tử của mảng

} list; //kieu danh sach list

**//tạo danh sách rỗng**

void init(list \*l){

(\*l).count=0;

}

**// nhập danh sách**

void Input(list \*l){

printf("nhap so phan tu danh sach: ");

scanf("%d", &(\*l).count);

int i;

for(i=0;i<(\*l).count;i++)

{

printf("\nnhap pt thu %d: ",i+1);

scanf("%d", &(\*l).a[i]);

}

}

**//xuất danh sách**

void Output(list l){

printf("\ndanh sach la: ");

int i;

for(i=0;i<l.count;i++){

printf("\t%d", l.a[i]);

}

}

**//kiem tra danh sach rong**

int isEmpty(list l){

return(l.count==0);

}

int isFull(list l){

return(l.count==0);

}

**// thêm phần tử x vào vị trí k**

int insert(list \*l, int x, int k){

if(isFull(\*l)){

printf("\nds day");

return 0;

}

if(k<1||k>(\*l).count+1){

printf("vi tri chen khong hop le");

return 0;

}

printf("nhap x: ");

scanf("%d", &x);

//di chuyen cac phan tu ve cuoi danh sach

for(int i=(\*l).count;i>=k-1;i--){

(\*l).a[i]=(\*l).a[i-1];

}

(\*l).a[k-1]=x; //chen x vao vi tri k

(\*l).count++;

return 1; }

// **tim kiem phan tu**

int search(list l,int x){ //ham search tra ve gia tri vi tri

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

if(l.a[i]==x)

{

return i+1; //return lai vi tri, vitri lech so voi i 1 don vi

}

return 0;

}

void timkiem(list l,int x){//ham in ra vi tri

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

{

if(l.a[i]==x)

{

printf("\ntim thay phan tu %d tai vi tri ", x);

printf("\t%d",i+1);

}

}

}

**// xoa 1 phan tu o vi tri k**

int del\_k(list \*l, int \*x, int k){

if(isEmpty(\*l)){

printf("\nds rong");

return 0;

}

if(k<=0||k>(\*l).count)

{

printf("vi tri xoa k hop le");

return 0;

}

\*x=(\*l).a[k-1]; // luu lai gia tri phan tu can xoa

//doofn cac phan tu ve truoc

for(int i=k-1;i<(\*l).count-1;i++)

{

(\*l).a[i]=(\*l).a[i+1];

}

(\*l).count--;

return 1;

}

**// xoa phan tu co gia tri x**

int del\_x(list \*l, int x){

if(isEmpty(\*l)){

printf("\nds rong");

return 0;

}

int i=search(\*l,x);

if(!i){

printf("ds khong co %d", x);

return 0;

}

do

{

del\_k(l,&x,i);

i=search(\*l,x);

}

while(i);

return 1;

}

**// dem so phan tu chan le am duong**

void demptchan(list l, int dem){

dem=0;

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

if(l.a[i]%2==0) //dem pt le thi l.a[i]%2!=0

dem++;

}

printf("\nso phan tu chan la %d", dem);

}

int demsoam(list l,int dem){

dem=0;

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

if(l.a[i]<0) //dem pt duong thi l.a[i]>0

{

dem++;

}

}

printf("\nso phan tu am la %d", dem);

return dem;

}

**//tính trung bình số âm**

void averageam(list l, float average){

int tong =0;

int dem=0;

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

if(l.a[i]<0) //dem pt duong thi l.a[i]>0

{

tong=tong+l.a[i];

dem=dem+1;

}

}

average=(float)tong/dem;

printf("\ntrung binh cong cac phan tu am la %.2f", average);

}

**//tính trung bình các số trong mảng**

void trungbinh(list l, float tb){

int tong =0;

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

tong=tong+l.a[i];

}

tb=(float)tong/l.count;

printf("\ntrung binh cong cac phan tu trong mang la %.2f", tb);

}

int main() {

list l;

init(&l); Input(&l); Output(l);

**// chen x vao vi tri k**

int x, k;

printf("\nnhap vi tri can them: ");

scanf("%d", &k);

if(insert(&l,x,k))

{ printf("\nds sau chen: ");

Output(l); }

**// xoa phan tu o vi tri b**

int b, p;

printf("\nnhap vi tri can xoa: ");

scanf("%d", &b);

if (del\_k(&l, &p, b))

{ printf("ds sau xoa: ");

Output(l); }

**// xoa phan tu co gia tri q**

int q;

printf("\nnhap phan tu can xoa: ");

scanf("%d", &q);

if(del\_x(&l,q))

{

printf("ds sau xoa: ");

Output(l);

}

**// tim phan tu**

int m;

printf("\nnhap phan tu can tim: ");

scanf("%d", &m);

int i=search(l,m);

if(i==0) printf("k tim thay %d trong ds", m);

else timkiem(l,m);

**//dem phan tu chan**

int dem;

demptchan(l, dem);

**//dem va tinh tb cong cac phan tu am**

int demam;

demsoam(l,demam);

float average;

averageam(l,average);

**//trung binh cong**

float tb;

trungbinh(l,tb);

return 0;

}

**Danh sách số nguyên cài bằng con trỏ**

#include<stdio.h> #include<conio.h> #include<stdlib.h>

typedef int item;

struct node{ **//xay dung 1 node trong danh sach**

item data;

node \*next; //next la con tro, tro den node tiep theo

};

typedef node \*list; //**list la 1 danh sach cac node**

**// tao ds rong**

void init(list &l){

l=NULL;

}

**// kiem tra ds rong**

int isEmpty(list l){

return(l==NULL);

}

**//tinh do dai ds**

int len(list l){

node \*p=l;

int i=0;

while(p!=NULL){

i++;

p=p->next;

} return i; }

**// tao 1 node trong ds**

node \*make\_node(node \*p, item x){//tao node p chua thong tin x

p= (node\*) malloc(sizeof(node));

p->next=NULL;

p->data=x;

return p; }

**// chen pt vao vi tri dau tien**

void insert\_first(list &l, item x){

node \*p;

p=make\_node(p,x);//tao node p chua x

p->next=l;

l=p;

}

**//chen node p vao vitri k trong danh sach**

void insert\_k(list &l, item x, int k){

node \*p, \*q=l;

int i=1;

if(k<1||k>len(l)+1)

printf("vi tri chen k hop le");

else

{ p=make\_node(p,x);

if(k==1)

insert\_first(l,x);

else

{

while(q!=NULL&& i!=k-1)//duyet den vi tri k-1

{

i++;

q=q->next;

}

//khi den vi tri k

p->next=q->next;

q->next=p;

}

}

}

**// chen vao cuoi danh sach neu p>len(L)**

void insert\_last(list &l, item x){

node \*p, \*q=l;

while(q->next!=NULL){

q=q->next;

}

q->next=p;

p->next=NULL;

}

**//nhap ds**

void input(list &l){

int i=0;

item x;

do{

i++;

printf("nhap phan tu thu %d: ",i);

scanf("%d", &x);

if(x!=0)

insert\_k(l,x,len(l)+1);

}while(x!=0);

}

**// xuat ds**

void ouput(list l){

node \*p=l;

while(p!=NULL)

{

printf("%5d", p->data);

p=p->next;

}

printf("\n"); }

**//tim phan tu co gia tri x**

int search(list l, item x){

node \*p=l;

int i=1;

while(p!=NULL&&p->data!=x)

{

p=p->next;

i++;

}

if(p!=NULL)

return i; //tra ve gia tri tim thay

else return 0;

}

**//xoa phan tu dau tien**

void del\_first(list &l, item &x){

x=l->data;//lay gia tri dau tien neu can dung

l=l->next;

}

**//xoa phan tu o vi tri k**

void del\_k(list &l, item &x, int k){

node \*p=l;

int i=1;

if(k<1||k>len(l))

printf("vi tri xoa k hop le");

else

{

if(k==1)

del\_first(l,x);

else

{

while(p!=NULL&& i!=k-1)

{ p=p->next;

i++;

}

p->next=p->next->next; //cho p tro sang vi tri sau vi tri k

}

}

}

**//xoa phan tu co gia tri x**

void del\_x (list &l, item x) //xoa phan tu x trong danh sach

{

while (search(l,x)) del\_k (l,x,search(l,x)); //trong khi van tim thay x thi van xoa

}

**// dem phan tu chan le am duong**

int count\_am(list l, int dem,float average){

printf("so cac pt am cua danh sach la: ");

node \*p=l;

dem=0;

int tong=0;

while(p!=NULL)

{

if(p->data<0)

{

tong=tong+p->data;

dem++;

}

p=p->next;

}

if(dem>0)

printf("%d",dem);

if(tong!=0);

average=(float)tong/dem;

printf("\ntbc so am la %.2f",average);

// return dem;

}

int main() {

list l;

init(l);

input(l);

ouput(l);

printf("chieu dai danh sach la %d", len(l));

**// tim phan tu co gia tri x**

item x;

printf("\nmoi nhap gia tri can tim:");

scanf("%d",&x);

int i=search(l,x);

if(i==0)

printf("\nds k co %d ", x);

else

printf("tim thay %d o vi tri %d",x,i);

**//them ptu x vao ds**

int k, a;

printf("\nmoi nhap vi tri chen:");

scanf("%d",&k);

printf("\nmoi nhap gia tri can chen:");

scanf("%d",&a);

insert\_k(l, a, k);

printf("\nds sau chen: ");

ouput(l);

**//xoa 1 ptu o vi tri k ra khoi danh sach**

int h, t;

printf("\nmoi nhap vi tri can xoa:");

scanf("%d",&h);

del\_k(l,t,h);

printf("\ndanh sach sau xoa");

ouput(l);

**//xoa 1 ptu co gia tri x ra khoi danh sach**

item m;

printf("\nmoi nhap gia tri can xoa:");

scanf("%d",&m);

del\_x(l,m);

ouput(l);

**//dem so ptu chan,le,am,duong,tbc so am**

int dem;

float average;

count\_am(l, dem, average);

return 0;

}