**Bài số 1: Tìm kiếm, sắp xếp**

#include<stdio.h>

#include<conio.h>

#include<stdlib.h>

int a[100],b[100], c[100],n;

void hoanvi(int &x, int &y)

{

int t=x;

x=y;

y=t;

}

void nhap(int a[],int n)

{

for(int i=1;i<=n;i++)

{

printf("a[%d]=",i);

scanf("%d",&a[i]);

}

}

void xuat(int a[],int n)

{

for(int i=1;i<=n;i++)

printf("%4d",a[i]);

}

int LinearSearch(int a[], int n, int x)

{ int i;

for (i=1; (a[i]!=x); i++);

if (i<=n)

return i; // a[i] la phan tu co khoa x

return -1; // khong tim thay phan tu co khoa x

}

int BinarySearch(int a[],int n,int x)

{

int left =1, right = n, midle;

while (left <= right)

{

midle = (left + right)/2;

if (x == a[midle])

return midle;//Tim thay tai vi tri mid

if (x<a[midle])right = midle -1;

else left = midle +1;

}

return -1; // khong tim thay phan tu co khoa x

}

void selectionsort(int a[],int n)

{

int min;

for(int i=1;i<n;i++)

{

min=i;

for(int j=i+1;j<=n;j++)

if(a[j]<a[min])

min=j;

hoanvi(a[min],a[i]);

}

}

void insertionsort(int a[],int n)

{

int pos;

int x;

for(int i=1;i<=n;i++)

{

x=a[i];pos=i-1;

while((pos>=0)&&(a[pos]>x))

{

a[pos+1]=a[pos];

pos--;

}

a[pos+1]=x;

}

}

void interchangeSort(int a[],int n)

{

int i,j;

for(i=1;i<n-1;i++)

for(j=i+1;j<n;j++)

if(a[j]<a[i])

hoanvi(a[i],a[j]);

}

void BubleSort(int a[],int n)

{

int i,j;

for(i=1;i<n;i++)

for(j=n;j>i;j--)

if(a[j]<a[j-1])

hoanvi(a[j],a[j-1]);

}

void ShakeSort(int a[],int n)

{

int i,j;

int left,right,k;

left=0;right=n-1;k=n-1;

while(left<right)

{

for(j=right;j>left;j--)

{

if(a[j]<a[j-1])

{

hoanvi(a[j],a[j-1]);

k=j;

}

}

left=k;

for(j=left;j<right;j++)

{

if(a[j]>a[j+1])

{

hoanvi(a[j],a[j-1]);

k=j;

}

}

right=k;

}

}

void Shift (int a[], int left, int right)

{

int x, curr, joint;

curr = left; joint =2\*curr;

x = a[curr];

while (joint <= right)

{

if (joint < right)

if (a[joint] < a[joint+1])

joint = joint+1;

if (a[joint]<x) break;

else

{

a[curr] = a[joint];

curr = joint;

joint = 2\*curr;

}

a[curr] = x;

}

}

void CreateHeap(int a[], int N)

{

int left;

for (left = (N)/2; left >= 1; left --)

Shift(a, left, N);

}

void Heapsort (int a[], int N)

{

int right;

CreateHeap(a, N);

# right = N;

while (right > 1)

{

hoanvi(a[1],a[right]);

right --;

Shift(a,1,right);

}

}

void Sellsort(int a[],int n)

{

int h[]={5,3,1};

int k=3;

int step,i,j;

int x,len;

for(step = 0;step<k;step++)

{

len=h[step];

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

{

x=a[i];

j=i-len;

while((x<a[j])&&(j>=1))

{

a[j+len]=a[j];

j=j-len;

}

a[j+len]=x;

}

}

}

void Quicksort(int a[],int l,int r)

{

int i,j;

int x;

x=a[(l+r)/2];

i=l;j=r;

do

{

while(a[i]<x) i++;

while(a[j]>x) j--;

if(i<=j)

{

hoanvi(a[i],a[j]);

i++;j--;

}

}while(i<j);

if(l<j)

Quicksort(a,l,j);

if(i<r)

Quicksort(a,i,r);

}

int min(int a,int b)

{

return a<b?a:b;

}

void Merge(int a[], int nb, int nc, int k)

{ int p, pb, pc, ib, ic, kb, kc;

p = pb = pc = 0; ib = ic = 0;

while((0 < nb)&&(0 < nc))

{

kb = min(k, nb); kc = min(k, nc);

if(b[pb+ib] <= c[pc+ic])

{

a[p++] = b[pb+ib]; ib++;

if(ib == kb)

{

for(; ic<kc; ic++)

a[p++] = c[pc+ic];

pb += kb; pc += kc; ib = ic = 0;

nb -= kb; nc -= kc;

}

}

else

{

a[p++] = c[pc+ic]; ic++;

if(ic == kc)

{

for(; ib<kb; ib++)

a[p++] = b[pb+ib];

pb += kb; pc += kc; ib = ic = 0;

nb -= kb; nc -= kc;

}

}

}

}

void mergesort(int a[], int n)

{

int p, pb, pc; // chi so tren cac mang a, b, c

int i, k = 1; // Do dai cua day con khi phan hoach

do

{

// tach a thanh b ,c

p = pb = pc = 0;

while(p < n)

{

for(i = 0; (p <=n)&&(i < k); i++)

b[pb++] = a[p++];

for(i = 0; (p <=n)&&(i < k); i++)

c[pc++] = a[p++];

}

Merge(a, pb, pc, k); //tron b, c lai thanh a

k \*= 2;

}

while(k <= n);

}

int main()

{ int x, vitri;

printf("\n nhap so phan tu:");

scanf("%d",&n);

nhap(a,n);

printf("\n day so da nhap la:");

xuat(a,n);

//Kiem tra ham tim tuan tu

printf("\n nhap khoa can tim:");

scanf("%d",&x);

vitri=LinearSearch(a, n, x);

if(vitri==-1)

printf("Khong tim thay %d",x);

else

printf("%d xuat hien tai vi tri %d",x,vitri);

/\*

//Kiem tra ham tim nhi phan

printf("\n nhap khoa can tim:");

scanf("%d",&x);

vitri=BinarySearch(a, n,x);

if(vitri==-1)

printf("Khong tim thay %d",x);

else

printf("%d xuat hien tai vi tri %d",x,vitri);

\*/

//Kiem tra ham selectionsort

//selectionsort(a,n);

//Kiem tra ham insertionsort

//insertionsort(a,n);

//Kiem tra ham interchangesort

//interchangesort(a,n);

//Kiem tra ham BubleSort

//BubleSort(a,n);

//Kiem tra ham ShakeSort

//ShakeSort(a,n);

//Kiem tra ham Heapsort

//Heapsort(a,n);

//Kiem tra ham Sellsort

//Sellsort(a,n);

//Kiem tra ham mergesort

// mergesort(a,n);

//Kiem tra ham Quicksort

//Quicksort(a,1,n);

//printf("\n day so sap xep la:");

//xuat(a,n);

//return 0;

getch();

}