**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();

}

**Bài số 2: Sắp xếp ngoài**

**Phương pháp trộn Run**

#include "stdio.h"

#include "conio.h"

int p,n;

void tao\_file(void)

{

//Tao file co n phan tu

int i,x;

FILE \*fp;

fp=fopen("D:\\Bang.txt","wb");

printf("Cho biet so phan tu : ");

scanf("%d", &n);

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

{

printf("Nhap so thu %d : ",i+1);

scanf("%d", &x);

fprintf(fp,"%3d",x);

}

fclose(fp);

}

void xuat\_file(void)

{

//Hien thi noi dung file len man hinh

int x;

FILE \*fp;

fp=fopen("D:\\Bang.txt","rb");

int i=0;

while(i<n)

{

fscanf(fp,"%d",&x);

printf("%3d",x);

i++;

}

fclose(fp);

}

void chia(FILE \*a, FILE \*b, FILE \*c, int p)

{

//Chia xoay vong file a cho file b va file c moi lan p phan tu cho

//den khi het file a.

int dem, x;

a=fopen("D:\\Bang.txt","rb");

b=fopen("D:\\Bang1.txt","wb");

c=fopen("D:\\Bang2.txt","wb");

while(!feof(a))

{

//Chia p phan tu cho b

dem=0;

while((dem<p) && (!feof(a)))

{

fscanf(a,"%3d",&x);

fprintf(b,"%3d",x);

dem++;

}

//Chia p phan tu cho c

dem=0;

while((dem<p) && (!feof(a)))

{

fscanf(a,"%3d",&x);

fprintf(c,"%3d",x);

dem++;

}

}

fclose(a);

fclose(b);

fclose(c);

}

void tron(FILE \*b, FILE \*c, FILE \*a, int p)

{

//Tron p phan tu tren b voi p phan tu tren c thanh 2\*p phan tu tren a cho den khi

//file b hoac c het

int stop, x, y, l, r;

a=fopen("D:\\Bang.txt","wb");

b=fopen("D:\\Bang1.txt","rb");

c=fopen("D:\\Bang2.txt","rb");

while((!feof(b)) && (!feof(c)))

{

l=0; //so phan tu cua b da ghi het len a

r=0; //so phan tu cua c da ghi het len a

fscanf(b,"%3d",&x);

fscanf(c,"%3d",&y);

stop=0;

while((l!=p) && (r!=p) && (!stop))

{

if(x<y)

{

fprintf(a,"%3d",x);

l++;

if((l<p) && (!feof(b)))

fscanf(b,"%3d",&x); //chua du p phan tu va chua het file b

else

{

fprintf(a,"%3d",y);

r++;

if((feof(b)))

stop=1;

}

}

else

{

fprintf(a,"%3d",y);

r++;

if((r<p) && (!feof(c)))

fscanf(c,"%3d",&y); //chua du p phan tu va chua het file c

else

{

fprintf(a,"%3d",x);

l++;

if((feof(c)))

stop=1;

}

}

}

//chep phan tu con lai cua p phan tu tren b len a

while((!feof(b)) && (l<p))

{

fscanf(b,"%3d",&x);

fprintf(a,"%3d",x);

l++;

}

//chep phan tu con lai cua p phan tu tren c len a

while((!feof(c)) && (r<p))

{

fscanf(c,"%3d",&y);

fprintf(a,"%3d",y);

r++;

}

}

if(!feof(b))

{

//chep phan tu con lai cua b len a

while(!feof(b))

{

fscanf(b,"%3d",&x);

fprintf(a,"%3d",x);

}

}

if(!feof(c))

{

//chep phan tu con lai cua c len a

while(!feof(c))

{

fscanf(c,"%3d",&x);

fprintf(a,"%3d",x);

}

}

fclose(a);

fclose(b);

fclose(c);

}

int main(void)

{

FILE \*a, \*b, \*c;

tao\_file();

printf("\n Tap tin nhap:");

xuat\_file();

p=1;

while(p<n)

{

chia(a,b,c,p);

tron(b,c,a,p);

p=2\*p;

}

printf("\n Tap tin da sap xep:");

xuat\_file();

getch();

}

**Phương pháp trộn tự nhiên**

#include <stdio.h>

#include <stdlib.h>

#include <conio.h>

#include <math.h>

#include <time.h>

#include <ctype.h>

typedef int DataType;

FILE \*F0,\*F1,\*F2;

int M,N,Eor;

/\*

Bien eor dung de kiem tra ket thuc Run hoac File

\*/

DataType X1,X2,X,Y;

void CreatFile(FILE \*Ft,int Num)

/\*Tao file co ngau nhien n phan tu\* \*/

{

srand(time(NULL));

Ft=fopen("e:\\bang.txt","wb");

for( int i = 0 ; i < Num ; i++)

{

X = rand()%(30);

fprintf(Ft,"%3d",X);

}

fclose(Ft);

}

void ListFile(FILE \*Ft)

/\*Hien thi noi dung cua file len man hinh \*/

{

DataType X,I=0;

Ft = fopen("e:\\bang.txt","rb");

while ( !feof(Ft) )

{

fscanf(Ft,"%3d",&X);

printf("%3d",X);

I++;

}

printf("\n\n");

fclose(Ft);

}

/\*\*/

void Copy(FILE \*Fi,FILE \*Fj)

{

//Doc phan tu X tu Tap tin Fi, ghi X vao Fj

//Eor==1, Neu het Run(tren Fi) hoac het File Fi

fscanf(Fi,"%3d",&X);

fprintf(Fj,"%3d",X);

if( !feof(Fi) )

{

fscanf(Fi,"%3d",&Y);

long curpos = ftell(Fi)-2;

fseek(Fi, curpos, SEEK\_SET);

}

if ( feof(Fi) ) Eor = 1;

else Eor = (X > Y) ? 1 : 0 ;

}

void CopyRun(FILE \*Fi,FILE \*Fj)

/\*Chep 1 Run tu Fi vao Fj \*/

{

do

Copy(Fi,Fj);

while ( !Eor);

}

void Distribute()

/\*Phan bo luan phien cac Run tu nhien tu F0 vao F1 va F2\*/

{

do

{

CopyRun(F0,F1);

if( !feof(F0) ) CopyRun(F0,F2);

}while( !feof(F0) );

fclose(F0);

fclose(F1);

fclose(F2);

}

void MergeRun()

/\*Tron 1 Run cua F1 va F2 vao F0\*/

{

do

{

fscanf(F1,"%3d",&X1);

long curpos = ftell(F1)-2;

fseek(F1, curpos, SEEK\_SET);

fscanf(F2,"%3d",&X2);

curpos = ftell(F2)-2;

fseek(F2, curpos, SEEK\_SET);

if( X1 <= X2 )

{

Copy(F1,F0);

if (Eor) CopyRun(F2,F0);

}

else

{

Copy(F2,F0);

if ( Eor ) CopyRun(F1,F0);

}

} while ( !Eor );

}

void Merge()

/\*Tron cac run tu F1 va F2 vao F0\*/

{

while( (!feof(F1)) && (!feof(F2)) )

{

MergeRun();

M++;

}

while( !feof(F1) )

{

CopyRun(F1,F0);

M++;

}

while( !feof(F2) )

{

CopyRun(F2,F0);

M++;

}

fclose(F0);

fclose(F1);

fclose(F2);

}

//Ham main

int main()

{

printf(" Nhap so phan tu: ");

scanf("%d",&N);

CreatFile(F0,N);

printf("\n Noi dung file goc:");

ListFile(F0);

do

{

F0=fopen("e:\\bang.txt","rb");

F1=fopen("e:\\bang1.txt","wb");

F2=fopen("e:\\bang2.txt","wb");

Distribute();

F0=fopen("e:\\bang.txt","wb");

F1=fopen("e:\\bang1.txt","rb");

F2=fopen("e:\\bang2.txt","rb");

M=0;

Merge();

}while (M != 1);

printf("\n Noi dung file da sap xep:");

ListFile(F0);

getch();

return 0;

}