BÀI THỰC HÀNH SỐ 4

Nội dung:

- Thao tác trên file

- Hàm đệ qui

**Bài 4.1: Xử lý số lớn**

**a) Nhập vào 2 số lớn (dùng chuỗi để lưu trữ) và ghi vào file Lab4\_1.inp**

**b) Tính và ghi vào file Lab4\_1.out giá trị tổng của 2 số lớn đó**

#include<stdio.h>

#include<string.h>

#include<stdlib.h>

int main()

{

FILE\* fi=fopen("E:\\baith4\\Lap4\_1.inp","w+");

FILE\* fo=fopen("E:\\baith4\\Lap4\_1.out","w");

char a[50],b[50];

int t[51];

//nhap chuoi va ghi vao file de luu tru

printf("nhap chuoi so thu nhat: ");

gets(a);

fputs(a,fi);

fprintf(fi,"\n");

printf("nhap chuoi so thu hai: ");

gets(b);

fputs(b,fi);

//xu ly cho 2 xau bang nhau

strrev(a);

strrev(b);

int x=strlen(a);

int y=strlen(b);

while(x!=y)

{

if (x<y)

{

strcat(a,"0");

x++;

}

else {

strcat(b,"0");

y++;

}

}

strrev(a);

strrev(b);

//tinh tong 2 so lon

int nho=0,ai,bi,tong=0;

for(int i=x-1;i>=0;i--)

{

ai=a[i]-48;

bi=b[i]-48;

tong=ai+bi+nho;

if (tong>9)

{

printf("%d\t",tong);

tong=tong%10;

nho=1;

}

else nho=0;

t[i]=tong;

}

for(int i=0;i<x;i++) fprintf(fo,"%d",t[i]);

//dong file

fclose(fi);

fclose(fo);

}

**Bài 4.2: Sử dụng file nhị phân**

**a) Tạo cấu trúc sách gồm các thông tin: tên sách, tên tác giả, nhà xuất bản, năm xuất**

**bản, giá bìa.**

**b) Viết chương trình nhập vào n quyển sách, và ghi thông tin n quyển sách vào file**

**nhị phân Lab4\_2.inp**

**c) Đọc dữ liệu từ file Lab4\_2.inp, sắp xếp và in ra màn hình n quyển sách theo**

**chiều giảm dần của năm xuất bản.**

**d) Lấy và in ra màn hình thông tin quyển sách thứ i trong file Lab4\_2.inp (với i**

**được nhập vào từ bàn phím).**

#include<stdio.h>

#include<string.h>

#define max 100

struct sach

{

char TS[30],TG[50],NXB[50];

int NAM;

float GB;

};

int main()

{

struct sach tt,tt1,mang[max];

int n,k;

FILE \*fi=fopen("E:/Lap4\_2.inp","w+b");

//nhap thong tin n quyen sach va luu vao file fi

printf("nhap so luong sach: ");

scanf("%d",&n);

fflush(stdin);

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

{

printf("ten sach:");

gets(tt.TS);

printf("tac gia:");

gets(tt.TG);

printf("nam xuat ban:");

scanf("%d",&tt.NAM);

fflush(stdin);

printf("Nha xuat ban:");

gets(tt.NXB);

printf("gia bia:");

scanf("%f",&tt.GB);

fflush(stdin);

fwrite(&tt,sizeof(sach),1,fi);

}

//xuat thong tin quyen sach bat ky

printf("nhap quyen sach can lay thong tin: ");

scanf("%d",&k);fflush(stdin);

fseek(fi,sizeof(sach),SEEK\_SET);

fread(&tt1,sizeof(sach),k,fi);

printf("ten sach: %s\n ",tt1.TS);

printf("tac gia: %s\n ",tt1.TG);

printf("nam xuat ban: %d\n",tt1.NAM);

printf("nha xuat ban: %s\n",tt1.NXB);

printf("gia bia: %f",tt1.GB);

}

**Bài 4.3: Viết hàm đệ qui thực hiện các yêu cầu:**

**a) Tính S(n) = an**

**b) Tính S(n) = 1**

**2 + 2**

**2 + 3**

**2+ ... + n**

**2 với n > 0**

**c) In các chữ số đảo ngược trong số nguyên dương n (VD: 1234 => 4321)**

**d) Đếm số chữ số trong số nguyên dương n (VD: 1234 => 4)**

**e) Tìm chữ số có giá trị lớn nhất trong số nguyên dương n (VD: 1534 => 5)**

**f) Tìm ước số chung lớn nhất của 2 số nguyên theo thuật toán Euclide.**

**g) Đổi số nguyên dương n sang hệ nhị phân.**

**h) Tính tổng n phần tử của mảng a (a[0], a[1] ,..., a[n-1])**

**i) Đếm số lần xuất hiện của phần tử x trong mảng a(a[0], a[1] ,..., a[n-1])**

**j) Tìm phần tử có giá trị lớn nhất trong mảng a(a[0], a[1] ,..., a[n-1])**

**k) Vi trùng cứ 1 giờ lại nhân đôi. Vậy sau n giờ sẽ có mấy con vi trùng nếu ban**

**đầu có 2 con?**

**l) Bài toán tháp Hà Nội: Có 3 cột A, B và C và cột A hiện có N đĩa. Tìm cách**

**chuyển N đĩa từ cột A sang cột C sao cho:**

**o Một lần chuyển 1 đĩa**

**o Đĩa lớn hơn phải nằm dưới**

**o Có thể sử dụng các cột A, B, C làm cột trung gian**

#include<stdio.h>

#include<math.h>

#define MAX 100

int max=0,t,dem=0;

int luythua(int x,int a);//cau a

int tongbp(int x);//cau b

void daonguoc(int x);//cau c

int demchuso(int x);//cau d

int timchusomax(int x);//cau e

int ucln(int x,int y);//cau f

void nhiphan(int x);//cau g

int tongmangb(int b[],int x);//cau h

void demx(int b[],int x,int y);//cau i

int ptmaxtrongmang(int b[],int x);//cau j

int vitrung(int x);//cau k

bool kt(int x,int y);

int main()

{

int n,a,cauc,max=0,b[MAX],caui;

printf("nhap n: ");

scanf("%d",&n);

printf("nhap a: ");

scanf("%d",&a);

printf("cau a\nluy thua %d mu %d la: %d\n",a,n,luythua(n,a));

printf("cau b\ntong binh phuong %d so dau tien la: %d\n",n,tongbp(n));

printf("cau c\nnhap so can dao nguoc: ");

scanf("%d",&cauc);

printf("so sau khi dao nguoc la: ");

daonguoc(cauc);

printf("\ncau d\nso chu so cua %d la: %d\n",cauc,demchuso(cauc));

printf("cau e\nchu so max trong %d la %d\n",cauc,timchusomax(cauc));

printf("cau f\nuoc chung lon nhat cua %d va %d la %d\n",a,n,ucln(a,n));

printf("cau g\nday nhi phan ung voi %d la: ",n);

nhiphan(n);

printf("\ncau h\n");

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

{

printf("nhap b[%d]",i);

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

}

printf("tong cua mang a la: %d",tongmangb(b,n-1));

printf("\ncau i\nnhap so can dem: ");scanf("%d",&caui);

demx(b,n-1,caui);

printf("so luong phan tu %d co trong mang la: %d",caui,dem);

printf("\ncau j\nso lon nhat trong mang la: %d",ptmaxtrongmang(b,n-1));

printf("\ncau k\nso vi trung sau %d gio la: %d",n,vitrung(n));

}

int luythua(int x,int a)

{

if (x==0) return 1;

else return luythua(x-1,a)\*a;

}

int tongbp(int x)

{

if (x==1) return 1;

else return tongbp(x-1)+pow(x,2);

}

void daonguoc(int x)

{

if (x<10) printf("%d",x);

else

{

printf("%d",x%10);

daonguoc(x/10);

}

}

int demchuso(int x)

{

if(x<10) return 1;

else

{

x=x/10;

return demchuso(x)+1;

}

}

int timchusomax(int x)

{

if(x<10) return max=x;

else

{

timchusomax(x/10);

if(max<x%10) return max=x%10;

}

}

int ucln(int x,int y)

{

if (y==0) return x;

else ucln(y,x%y);

}

void nhiphan(int x)

{

if(x==1) printf("%d",x);

else

{

nhiphan(x/2);

printf("%d",x%2);

}

}

int tongmangb(int b[],int x)

{

if (x==0) return b[0];

else return tongmangb(b,x-1)+b[x];

}

void demx(int b[],int x,int y)

{

if(x>=0)

{

if(kt(b[x],y)==true) dem++;

demx(b,x-1,y);

}

}

int ptmaxtrongmang(int b[],int x)

{

if (x==0) return max=b[x];

else

{

ptmaxtrongmang(b,x-1);

if (b[x]>max) return max=b[x];

}

}

int vitrung(int x)

{

if (x==0) return 2;

else return vitrung(x-1)\*2;

}

bool kt(int x,int y)

{

if(x==y) return true;

else return false;

}