BÀI THỰC HÀNH SỐ 1

Nội dung:

- Các nguyên tắc lập trình

- Tổ chức, tối ưu mã lệnh

- Xử lý lỗi và ngoại lệ

- Xây dựng hàm**Bài 1.1. Viết chương trình giải và biện luận phương trình bậc 2: ax2 + bx + c = 0.**#include<stdio.h>

#include<math.h>

float a,b,c,d;

int main()

{

printf("nhap he so a ");

scanf("%f",&a);

printf("nhap he so b ");

scanf("%f",&b);

printf("nhap he so c ");

scanf("%f",&c);

if(a==0)

if(b==0)

if(c==0) printf("phuong trinh co vo so nghiem");

else printf("phuong trinh vo nghiem");

else printf("phuong trinh co nghiem duy nhat %5.5f",float(-c)/b);

else

{

d=pow(b,2)-4\*a\*c;

if (d==0) printf("phuong trinh co nghiem kep la: %5.5f",float(-b)/(2\*a));

else

if (d<0) printf("phuong trinh vo nghiem");

else

{

float x1=(-b+sqrt(d))/(2\*a);

float x2=(-b-sqrt(d))/(2\*a);

printf("phuong trinh co 2 nghiem phan biet \nx1= %5.5f \nx2= %5.5f",x1,x2);

}

}

}

**Bài 1.2. Viết chương trình nhập vào 2 số nguyên tương ứng với tử số (ts) và mẫu số của một phân số, xuất ra màn hình phân số ở dạng tối giản.**

#include<stdio.h>

int a,b;

int ucln(int a,int b);

int main()

{

printf("tu so: ");

scanf("%d",&a);

printf("mau so: ");

scanf("%d",&b);

if (b==0) printf("khong ton tai phan so");

else if (a==0) printf("phan so = 0");

else if (a==b)printf("phan so = 1");

else

{

printf("phan so: %d/%d",a,b);

printf("\nphan so toi gian: %d/%d",a/ucln(a,b),b/ucln(a,b));

}

}

int ucln(int a,int b)

{

int t=b%a;

while (t!=0)

{

t= a%b;

a= b;

b= t;

}

return a;

}

**Bài 1.3. Viết chương trình thực hiện các yêu cầu sau:**

**- Nhập số nguyên dương n gồm k chữ số (0 <k ≤ 5) , đếm xem n có bao nhiêu chữ**

**số là số nguyên tố.**

**- Nhập vào số nguyên dương n ( n ≤ 100), in ra tất cả các số nguyên tố nhỏ hơn n**

**(sử dụng thuật toán sàng Eratosthenes)**

#include<stdio.h>

#include<math.h>

int ktsnt(int x);

int main()

{

int n,dem1=0;

printf("nhap n: ");

scanf("%d",&n);

while(n/10!=0)

{

int m=n%10;

if ((m==2)||(m==3)||(m==5)||(m==7)) dem1++;

n=n/10;

}

if ((n==2)||(n==3)||(n==5)||(n==7)) dem1++;

printf("so chu so nguyen to la %d",dem1);

}

int ktsnt(int x)

{

if (x<=1) return 0;

else if(x<4) return 1;

else

{

int dem=0;

int y=sqrt(x);

for(int i=2;i<=y;i++)

if(x%i==0)

{

dem++;

break;

}

if (dem==0) return 1;

else return 0;

}

}

**Bài 1.5: Cho biết chương trình dưới đây thực hiện yêu cầu gì? Thực hiện debug chương trình để xem kết quả của biến tong va sochuso tại lần lặp thứ i.**

#include<stdio.h>

int main()

{

long n, tong=0;

int sochuso=0;

float tb;

printf ("Nhap vao gia tri n gom k chu so") ;

scanf ("%ld",&n) ;

while(n > 0)

{

tong = tong + n%10 ; //tinh tong cac chu so cua n

sochuso++ ; //dem so chu so cua so n

n = n/10 ; //loai bo lan luot cac chu so

}

tb = 1.0\*tong/sochuso ;

printf ("Gia tri trung binh la: %f", tb) ;

}

BÀI THỰC HÀNH SỐ 2

Nội dung:

- Xây dựng hàm

- Hàm nâng cao

- Con trỏ cơ bản

**Bài 2.1. Dùng hàm main có đối số**

**a) Viết chương trình có tên Hello, nhận 1 đối số s và xuất ra màn hình “Chao cac ban lop s!”**

**b) Viết chương trình có tên TinhToan sao cho khi gõ: TinhToan 250 – 170 sẽ xuất ra màn hình 250 – 170 = 80 (có thể thay bằng +, \*, /)**

#include<stdio.h>

int main(int argc, char \*argv[])

int TinhToan(int a,int b,char c)

{

if (c=='+') return a+b;

if (c=='-') return a-b;

if (c=='\*') return a\*b;

if (c=='/') return a/b;

}

//a.

{

if (argc==2) printf("chao hoc sinh lop %s", argv[1]);

else printf("Err!");

return 0;

}

//b.

{

if (argc == 4)

{

int a = atoi (argv[1]);

int b = atoi (argv[3]);

char c= \*argv[2];

printf("%d %c %d = %d",a,c,b,TinhToan(a,b,c));

}

else

printf("Cu phap loi!VD: TinhToan 250 - 170 ");

}

**Bài 2.2: Viết chương trình nhập vào số kw điện, số m3 nước tiêu của một phòng trọ. Tính**

**và in ra màn hình số tiền phải trả: số kw \* 4 + số m3 \* 7 + tiền phòng, biết tiền phòng mặc**

**định là 1200.**

**Gợi ý: Dùng hàm có đối số có giá trị mặc định.**

#include<stdio.h>

int tongtien(int kw, int m3,int tphong=1200);

int main()

{

int kw,m3;

printf("nhap so kw dien: ");

scanf("%d",&kw);

printf("nhap so m3 nuoc: ");

scanf("%d",&m3);

printf("so tien phai tra la: %d",tongtien(kw,m3));

}

int tongtien(int kw, int m3,int tphong)

{

int tien;

tien=kw\*4+m3\*7+tphong;

return tien;

}

**Bài 2.3:**

**a) Viết và sử dụng Macro để tính tổng của hai số a, b.**

**b) Viết và sử dụng macro để hoán đổi giá trị giữa 2 phần tử a, b;**

**c) Viết và sử dụng hàm nội tuyến để tính tích của hai số a, b.**

**d) Viết hàm trả về tham chiếu đến phần tử của mảng, sử dụng hàm thực hiện các thao**

**tác nhập mảng gồm n số nguyên (3 ≤ n ≤ 50), xuất ra mảng được sắp xếp theo chiều**

**tăng dần.**

#include<stdio.h>

#define tong(a,b) (a+b)

#define hoandoi(a,b,c) (c=a,a=b,b=c)

inline int tich(int a, int b);

#define max 100

int A[max];

int sx(int x,int y)

{

int d,c,tg,g;

d=x;

c=y;

g=A[(d+c)/2];

do

{

while (A[d]<g) d++;

while (A[c]>g) c--;

if (d<=c)

{

tg=A[d];

A[d]=A[c];

A[c]=tg;

d++;

c--;

}

}

while (d<c);

if (d<y) sx(d,y);

if (c>x) sx(x,c);

}

int &V(int A[],int j)

{

return A[j-1];

}

int main()

{

{

int a,b,c;

printf("nhap a: ");

scanf("%d",&a);

printf("nhap b: ");

scanf("%d",&b);

printf("tong %d + %d = %d",a,b,tong(a,b));

hoandoi(a,b,c);

printf("\nsau khi hoan doi la a=%d b=%d:",a,b);

printf("tich %d \* %d = %d",a,b,tich(a,b));

}

inline int tich(int a, int b)

{

int t=a\*b;

return t;

}

{

int n,i;

do

{

printf("nhap n");

scanf("%d",&n);

}

while ((n<=3)||(n>=50));

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

{

printf("nhap phan tu A[%d]",i);

scanf("%d",&V(A,i));

}

sx(0,n-1);

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

printf("%3d",A[i-1]);

}

}

**Bài 2.4: Tạo cấu trúc học phần gồm: mã học phần, tên học phần, số đơn vị học trình. Viết**

**chương trình thực hiện:**

**c) Khai báo biến con trỏ trỏ tới một biến kiểu cấu trúc học phần.**

**d) Nhập giá trị cho các thành phần của cấu trúc.**

**e) Xuất thông tin của học phần ra màn hình trên 1 dòng.**

#include<stdio.h>

#include<stdlib.h>

struct HocPhan

{

char MHP[6];

char THP[15];

int TC;

};

int main()

{

struct HocPhan hp, \*Hp=&hp;

printf("Nhap du lieu cho hp:\n");

printf("MHP: "); fflush(stdin);

gets(hp.MHP);

printf("ten hoc phan: "); fflush(stdin);

gets(hp.THP);

printf("so tin chi: "); fflush(stdin);

scanf("%d", &hp.TC);

printf("\n --------- Thong tin hoc phan -----\n");

printf("%-20s %-30s %-7s\n", "MHP", "Ten hoc phan", "so tin chi");

printf("%-20s %-30s %d\n", Hp->MHP, Hp->THP, hp.TC);

}

**Bài 2.5: Viết chương trình thực hiện các yêu cầu sau (dùng con trỏ để cấp phát cùng nhớ**

**cho mảng, ma trận, dùng macro cho vòng lặp for)**

**a) Thao tác trên mảng 1 chiều:**

**- Nhập vào mảng gồm n số nguyên ( 4 < n < 50)**

**- Xuất ra mảng vừa nhập**

**- Tìm và in ra màn hình tất cả các cặp số nguyên tố cùng nhau trong mảng**

#include<stdio.h>

#include<stdlib.h>

#include<math.h>

#define for(i,a,b) for(int i=a;i<=b;i++)

int ktsnt(int x);

int main()

{

int n,j,k;

do

{

printf("nhap n: ");

scanf("%d",&n);

}

while ((n<=4)||(n>=50));

int \*p=(int\*) calloc(n,sizeof(int));

for(i,0,n-1)

{

printf("nhap phan tu thu %d: ",i+1);

scanf("%d",p+i);

}

printf("Manh vua nhap la:\n");

for(i,0,n-1) printf("%3d",\*(p+i));

printf("\ncac cap so nguyen to trong mang la:\n");

for(i,0,n-1)

if(ktsnt(\*(p+i))==1)

for(j,i+1,n-1)

if(ktsnt(\*(p+j))==1)printf("%d\t%d\n",\*(p+i),\*(p+j));

free(p);

}

int ktsnt(int x)

{

if (x<=1) return 0;

else if(x<4) return 1;

else

{

int dem=0;

int y=sqrt(x);

for(k,2,y)

if(x%k==0)

{

dem++;

break;

}

if (dem==0) return 1;

else return 0;

}

}

**b) Thao tác trên mảng 2 chiều:**

**- Nhập vào ma trận gồm n x m số nguyên ( 2 ≤ n, m ≤ 10)**

**- Xuất ra ma trận vừa nhập**

**- Tính và in ra màn hình trung bình cộng các phần tử trong ma trận.**

#include<stdio.h>

#include<stdlib.h>

#include<math.h>

#define for(i,a,b) for(int i=a;i<b;i++)

int main()

{

int n,m,j;

int tong=0;

do

{

printf("nhap dong: ");

scanf("%d",&n);

printf("nhap cot: ");

scanf("%d",&m);

}

while ((n<2)||(n>10)||(m<2)||(m>10));

int spt=m\*n;

//khoi tao bo cap phat dong cho ma tran

int \*\*a;

a = new int \*[n];

for (i,0,n)

a[i] = new int[m];

//nhap ma tran

for (i,0,n)

for (j,0,m)

{

printf("nhap phan tu a[%d][%d]: ",i,j);

scanf("%d",&a[i\*m+j]);

}

// in ma tran vua nhap

for (i,0,n)

{

for (j,0,m)

printf("%d\t",(a[i\*m+j]));

printf("\n");

}

//tinh tbc cac phan tu tran ma tran

for (i,0,n)

for (j,0,m)

tong=tong+\*a[i\*m+j];

tong=tong/spt;

printf("\ntbc cac phan tu trong mang la: %d",tong);

}