#include <stdio.h>

#include <math.h>

struct PhanSo

{

int tu;

int mau;

};

void xuatPhanSo (PhanSo a);

void nhapPhanSo (PhanSo &a);

int gcd(int a, int b);

int lcm(int a, int b);

PhanSo rutGonPhanSo (PhanSo &a);

void quyDongPhanSo (PhanSo &a, PhanSo &b);

PhanSo tinhTong (PhanSo a, PhanSo b);

PhanSo tinhHieu (PhanSo a, PhanSo b);

PhanSo tinhTich (PhanSo a, PhanSo b);

PhanSo tinhThuong (PhanSo a, PhanSo b);

void soSanh (PhanSo a, PhanSo b);

void xuat (PhanSo kq);

void main ()

{

PhanSo a, b, t, h, s, th;

nhapPhanSo (a);

printf ("Phan So a: ");

xuatPhanSo (a);

nhapPhanSo (b);

printf ("Phan So b: ");

xuatPhanSo (b);

t = tinhTong (a, b);

printf ("Tong = ");

xuat(t);

h = tinhHieu (a, b);

printf ("Hieu = ");

xuat (h);

s = tinhTich (a, b);

printf ("Tich = ");

xuat (s);

th = tinhThuong (a, b);

printf ("Thuong = ");

xuat (th);

quyDongPhanSo (a, b);

soSanh (a, b);

}

void nhapPhanSo (PhanSo &a)

{

scanf ("%d%d", &a.tu, &a.mau);

}

void xuatPhanSo (PhanSo a)

{

printf ("%d/%d\n", a.tu, a.mau);

}

int gcd(int a, int b)

{

if (b == 0) return a;

return gcd(b, a % b);

}

int lcm(int a, int b)

{

return a\*b/gcd(a,b);

}

PhanSo rutGonPhanSo (PhanSo &a)

{

int GCD = gcd(a.tu, a.mau);

if (GCD ==1)

return a;

a.tu = a.tu/GCD;

a.mau = a.mau/GCD;

return a;

}

void quyDongPhanSo (PhanSo &a, PhanSo &b)

{

int LCM = lcm (a.mau, b.mau);

a.tu = a.tu\*(LCM/a.mau);

a.mau = LCM;

b.tu = b.tu\*(LCM/b.mau);

b.mau = LCM;

printf ("Quy dong: a=%d/%d b=%d/%d\n", a.tu, a.mau, b.tu, b.mau);

}

PhanSo tinhTong (PhanSo a, PhanSo b)

{

PhanSo kq;

kq.tu = a.tu\*b.mau + a.mau\*b.tu;

kq.mau = a.mau\*b.mau;

kq = rutGonPhanSo (kq);

return kq;

}

PhanSo tinhHieu (PhanSo a, PhanSo b)

{

PhanSo kq;

kq.tu = a.tu\*b.mau - a.mau\*b.tu;

kq.mau = a.mau\*b.mau;

kq = rutGonPhanSo (kq);

return kq;

}

PhanSo tinhTich (PhanSo a, PhanSo b)

{

PhanSo kq;

kq.tu = a.tu\*b.tu;

kq.mau = a.mau\*b.mau;

kq = rutGonPhanSo (kq);

return kq;

}

PhanSo tinhThuong (PhanSo a, PhanSo b)

{

PhanSo kq;

kq.tu = a.tu\*b.mau;

kq.mau = a.mau\*b.tu;

kq = rutGonPhanSo (kq);

return kq;

}

void soSanh (PhanSo a, PhanSo b)

{

printf ("So sanh: ");

a= rutGonPhanSo (a);

b = rutGonPhanSo (b);

if (a.tu == b.tu && a.mau == b.mau)

printf ("a=b");

else

if (a.tu\*b.mau > b.tu\*a.mau)

printf ("a>b");

printf ("a<b");

}

void xuat (PhanSo kq)

{

printf ("%d/%d\n", kq.tu, kq.mau);

}

#include <stdio.h>

#include <math.h>

//

struct PhanSo

{

int tu;

int mau;

};

//

int gcd (int a, int b);

int kiemTraHopLe (PhanSo a);

void nhapPhanSo (PhanSo &a);

void xuatPhanSo (PhanSo a);

void nhapMangPhanSo (PhanSo A[], int &n);

void xuatMangPhanSo (PhanSo A[], int n);

PhanSo rutGonPhanSo (PhanSo &a);

void rutGonMangPhanSo (PhanSo A[], int n);

PhanSo tinhTong (PhanSo a, PhanSo b);

PhanSo tinhTongMangPhanSo (PhanSo A[], int n);

int soSanh (PhanSo a, PhanSo b);

PhanSo timMax (PhanSo A[], int n);

PhanSo timMin (PhanSo A[], int n);

void sapXepMangTang (PhanSo A[], int n);

//

void main ()

{

int n;

PhanSo A[1000], kq, max, min;

nhapMangPhanSo (A, n);

printf ("Mang vua nhap: ");

xuatMangPhanSo (A, n);

rutGonMangPhanSo (A, n);

kq = tinhTongMangPhanSo (A, n);

printf ("\nTong cac phan so: ");

xuatPhanSo (kq);

max = timMax (A, n);

printf ("\nPhan so lon nhat: ");

xuatPhanSo (max);

min = timMin (A, n);

printf ("\nPhan so nho nhat: ");

xuatPhanSo (min);

sapXepMangTang (A, n);

printf ("\nMang sap xep: ");

xuatMangPhanSo (A, n);

}

// tim uoc so chung lon nhat

int gcd (int a, int b)

{

a = abs(a);

b = abs(b);

while (a!=b)

{

if (a>b)

a = a-b;

else

b = b-a;

}

return a;

}

// kiem tra phan so co mau bang 0

int kiemTraHopLe (PhanSo a)

{

int hl = 1;

if (a.mau ==0)

hl = 0;

return hl;

}

// nhap phan so

void nhapPhanSo (PhanSo &a)

{

do

{

scanf ("%d%d", &a.tu, &a.mau);

if (!kiemTraHopLe(a))

printf ("phan so khong hop le\nnhap lai: ");

}

while (!kiemTraHopLe(a));

}

// xuat phan so

void xuatPhanSo (PhanSo a)

{

if (a.mau == 1)

printf ("%d; ", a.tu);

else

if (a.mau == -1)

{

a.tu = a.tu \* -1;

a.mau = 1;

printf ("%d; ", a.tu);

}

else

if (a.tu == a. mau)

printf ("%d; ", 1);

else

printf ("%d/%d; ", a.tu, a.mau);

}

// nhap mang phan so

void nhapMangPhanSo (PhanSo A[], int &n)

{

scanf ("%d", &n);

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

nhapPhanSo(A[i]);

}

// xuat mang phan so

void xuatMangPhanSo (PhanSo A[], int n)

{

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

xuatPhanSo (A[i]);

}

// rut gon phan so

PhanSo rutGonPhanSo (PhanSo &a)

{

int GCD = gcd (a.tu, a.mau);

if (GCD==1)

return a;

else

{

a.tu = a.tu/GCD;

a.mau = a.mau/GCD;

}

return a;

if (a.mau <0)

{

a.tu = -1\*a.tu;

a.mau = -1\*a.mau;

}

return a;

}

// rut gon mang phan so

void rutGonMangPhanSo (PhanSo A[], int n)

{

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

rutGonPhanSo (A[i]);

}

// tinh tong hai phan so

PhanSo tinhTong (PhanSo a, PhanSo b)

{

PhanSo kq;

kq.tu = a.tu\*b.mau + a.mau\*b.tu;

kq.mau = a.mau\*b.mau;

kq = rutGonPhanSo (kq);

return kq;

}

// tinh tong mang phan so

PhanSo tinhTongMangPhanSo (PhanSo A[], int n)

{

PhanSo kq;

PhanSo t = A[0];

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

t = tinhTong (t, A[i]);

kq = rutGonPhanSo (t);

return kq;

}

// so sanh hai phan so

int soSanh (PhanSo a, PhanSo b)

{

a = rutGonPhanSo (a);

b = rutGonPhanSo (b);

if (a.tu == b.tu && a.mau==b.mau)

return 0;

else

if (a.tu\*b.mau > b.tu\*a.mau)

return 1;

return -1;

}

// tim phan so lon nhat

PhanSo timMax (PhanSo A[], int n)

{

PhanSo max;

max = A[0];

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

if (soSanh (A[i], max) == 1)

max = A[i];

return max;

}

// tim phan so nho nhat

PhanSo timMin (PhanSo A[], int n)

{

PhanSo min;

min = A[0];

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

if (soSanh (A[i], min) == -1)

min = A[i];

return min;

}

// sap mang phan so tang dan

void sapXepMangTang (PhanSo A[], int n)

{

PhanSo tam;

for (int i=0; i<n-1; i++)

{

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

{

if (soSanh (A[i], A[j]) == 1)

{

tam = A[i];

A[i] = A[j];

A[j] = tam;

}

}

}

}