**Phan So De**

#include <cmath>

#include <cstdio>

#include <vector>

#include <iostream>

#include <algorithm>

#define SO\_LUONG\_MANG 100

using namespace std;

class PhanSo

{

private:

int iTuSo;

int iMauSo;

int Gcd(int iTuSo, int iMauSo);

public:

PhanSo();

void Input();

void Output();

void Compact();

~PhanSo();

};

int PhanSo::Gcd(int iTuSo, int iMauSo)

{

iTuSo = abs(iTuSo);

iMauSo = abs(iMauSo);

if (iTuSo == 0 || iMauSo == 0)

return (iTuSo + iMauSo);

while (iTuSo != iMauSo)

{

if (iTuSo > iMauSo)

iTuSo = iTuSo - iMauSo;

else

iMauSo = iMauSo - iTuSo;

}

return iMauSo;

}

PhanSo::PhanSo()

{

iTuSo = 0;

iMauSo = 1;

}

PhanSo::~PhanSo()

{

}

void PhanSo::Input()

{

cin >> iTuSo;

do

{

cin >> iMauSo;

} while (iMauSo == 0);

}

void PhanSo::Output()

{

cout << iTuSo << "/" << iMauSo << endl;

}

void PhanSo::Compact()

{

int iUCLN = Gcd(iTuSo, iMauSo);

iTuSo = iTuSo / iUCLN;

iMauSo = iMauSo / iUCLN;

}

int main()

{

PhanSo psA[SO\_LUONG\_MANG];

int iSoLuong;

cin >> iSoLuong;

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

{

psA[i].Input();

psA[i].Compact();

}

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

{

psA[i].Output();

}

system("pause");

return 0;

}

**Phan So Nhung Kho Hon**

#include <cmath>

#include <cstdio>

#include <vector>

#include <iostream>

#include <algorithm>

#define SO\_LUONG\_MANG 100

using namespace std;

class PhanSo

{

private:

int iTuSo;

int iMauSo;

public:

PhanSo();

void Input();

void Output();

float Float();

~PhanSo();

};

PhanSo::PhanSo()

{

iTuSo = 0;

iMauSo = 1;

}

PhanSo::~PhanSo()

{

}

void PhanSo::Input()

{

cin >> iTuSo;

do

{

cin >> iMauSo;

} while (iMauSo == 0);

}

void PhanSo::Output()

{

cout << iTuSo << "/" << iMauSo << endl;

}

float PhanSo::Float()

{

return float(iTuSo) / iMauSo;

}

int main()

{

PhanSo psA[SO\_LUONG\_MANG];

int iSoLuong;

cin >> iSoLuong;

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

{

psA[i].Input();

}

int fMax = 0;

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

{

if (psA[i].Float() > psA[fMax].Float()) fMax = i;

}

int fMin = 0;

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

{

if (psA[i].Float() < psA[fMin].Float()) fMin = i;

}

psA[fMin].Output();

psA[fMax].Output();

system("pause");

return 0;

}

**Van La Phan So**

#include <cmath>

#include <cstdio>

#include <vector>

#include <iostream>

#include <algorithm>

#define SO\_LUONG\_MANG 100

using namespace std;

class PhanSo

{

private:

int iTuSo;

int iMauSo;

int Gcd(int iTuSo, int iMauSo);

public:

PhanSo();

void Input();

void Output();

PhanSo Add(PhanSo ps2);

PhanSo Subtract(PhanSo ps2);

PhanSo Multiply(PhanSo ps2);

PhanSo Divide(PhanSo ps2);

void Compact();

~PhanSo();

};

int PhanSo::Gcd(int iTuSo, int iMauSo)

{

iTuSo = abs(iTuSo);

iMauSo = abs(iMauSo);

if (iTuSo == 0 || iMauSo == 0)

return (iTuSo + iMauSo);

while (iTuSo != iMauSo)

{

if (iTuSo > iMauSo)

iTuSo = iTuSo - iMauSo;

else

iMauSo = iMauSo - iTuSo;

}

return iMauSo;

}

PhanSo::PhanSo()

{

iTuSo = 0;

iMauSo = 1;

}

PhanSo::~PhanSo()

{

}

void PhanSo::Input()

{

cin >> iTuSo;

do

{

cin >> iMauSo;

} while (iMauSo == 0);

}

void PhanSo::Output()

{

if(this-> iMauSo!=0)

{

cout << iTuSo << "/" << iMauSo << endl;

}

}

PhanSo PhanSo::Add(PhanSo ps2)

{

PhanSo psA;

psA.iTuSo = this->iTuSo \* ps2.iMauSo + this->iMauSo \* ps2.iTuSo;

psA.iMauSo = this->iMauSo \* ps2.iMauSo;

return psA;

}

PhanSo PhanSo::Subtract(PhanSo ps2)

{

PhanSo psS;

psS.iTuSo = this->iTuSo \* ps2.iMauSo - this->iMauSo \* ps2.iTuSo;

psS.iMauSo = this->iMauSo \* ps2.iMauSo;

return psS;

}

PhanSo PhanSo::Multiply(PhanSo ps2)

{

PhanSo psM;

psM.iTuSo = this->iTuSo \* ps2.iTuSo;

psM.iMauSo = this->iMauSo \* ps2.iMauSo;

return psM;

}

PhanSo PhanSo::Divide(PhanSo ps2)

{

PhanSo psD;

psD.iTuSo = this->iTuSo \* ps2.iMauSo;

psD.iMauSo = this->iMauSo \* ps2.iTuSo;

return psD;

}

void PhanSo::Compact()

{

int iUCLN = Gcd(iTuSo, iMauSo);

iTuSo = iTuSo / iUCLN;

iMauSo = iMauSo / iUCLN;

if (iMauSo < 0)

{

iTuSo \*= (-1);

iMauSo \*= (-1);

}

}

int main()

{

PhanSo ps1, ps2, psA, psS, psM, psD;

ps1.Input();

ps2.Input();

psA = ps1.Add(ps2);

psA.Compact();

psA.Output();

psS = ps1.Subtract(ps2);

psS.Compact();

psS.Output();

psM = ps1.Multiply(ps2);

psM.Compact();

psM.Output();

psD = ps1.Divide(ps2);

psD.Compact();

psD.Output();

system("pause");

return 0;

}

**Tim Ngay Ke Tiep**

#include <cmath>

#include <cstdio>

#include <vector>

#include <iostream>

#include <algorithm>

#include <iostream>

#include <iostream>

using namespace std;

int NamNhuan(int);

class Date

{

private:

int d;

int m;

int y;

public:

void Nhap();

void NextDay();

};

void Date::Nhap()

{

cin >> d >> m >> y;

}

void Date::NextDay()

{

int s = 0;

if (this->y < 0)

{

s = 1;

}

switch (this->m)

{

case 1: case 3: case 5: case 7: case 8: case 10:

{

if (this->d > 0 && this->d < 31)

{

this->d = this->d + 1;

s = 0;

break;

}

if (this->d == 31)

{

this->d = 1;

this->m = this->m + 1;

s = 0;

break;

}

}

case 12:

{

if (this->d > 0 && this->d < 31)

{

this->d = this->d + 1;

s = 0;

break;

}

if (this->d == 31)

{

this->d = 1;

this->m = 1;

this->y = this->y + 1;

s = 0;

break;

}

}

case 4: case 6: case 9: case 11:

{

if (this->d > 0 && this->d < 30)

{

this->d = this->d + 1;

s = 0;

break;

}

if (this->d == 30)

{

this->d = 1;

this->m = this->m + 1;

s = 0;

break;

}

}

case 2:

{

if (NamNhuan(this->y) == 1)

{

if (this->d > 0 && this->d < 29)

{

this->d = this->d + 1;

s = 0;

break;

}

if (this->d == 29)

{

this->d = 1;

this->m = this->m + 1;

s = 0;

break;

}

}

else

{

if (this->d > 0 && this->d < 28)

{

this->d = this->d + 1;

s = 0;

break;

}

if (this->d == 28)

{

this->d = 1;

this->m = this->m + 1;

s = 0;

break;

}

}

}

default:

{

s = 1;

break;

}

}

if (s == 0)

{

cout << d << "/" << m << "/" << y;

}

else

{

cout << "ERROR";

}

}

int NamNhuan(int a)

{

if (a % 400 == 0)

return 1;

if (a % 4 == 0 && a % 100 != 0)

return 1;

return 0;

}

int main()

{

Date a;

a.Nhap();

a.NextDay();

return 0;

}

**Class Hoc Sinh**

#include <cmath>

#include <cstdio>

#include <vector>

#include <iostream>

#include <algorithm>

#include <cstring>

using namespace std;

class Class

{

private:

string Name;

float dt, dv;

public:

void Input();

void Ouput();

};

void Class::Input()

{

getline(cin,this->Name);

do

{

cin>>this->dt;

cin>>this->dv;

}while((dt<0||dt>10)||(dv<0||dv>10));

}

void Class::Ouput()

{

cout << "Thong tin hoc sinh:" << endl;

cout <<"Ho ten: " << this->Name << endl;

cout << "Diem toan: " << this->dt << endl;

cout << "Diem van: " << this->dv << endl;

cout << "Diem trung binh: " << (this->dt+ this->dv)/2 << endl;

}

int main()

{

Class a;

a.Input();

a.Ouput();

return 0;

}