

Bài Tập Thực Hành Tuần 1

Môn Phương Pháp Lập Trình Hướng Đối Tượng

Chương trình 1:

```
#include<iostream>
#include<string>
using namespace std;

void Nhap(string&, int&, int&);
void XuLy(int, int, float&);
void Xuat(string, int, int, float);

int main()
{
    string ht;
    int t, v;
    float dtb;
    Nhap(ht, t, v);
    XuLy(t, v, dtb);
    Xuat(ht, t, v, dtb);
    return 1;
}

void Nhap(string& hoten, int& toan, int& van)
{
    cout << "Nhap ho ten: ";
    getline(cin, hoten);
    cout << "Nhap toan: ";
    cin >> toan;
    cout << "Nhap van: ";
    cin >> van;
}

void XuLy(int toan, int van, float& dtb)
{
    dtb = (float)(toan + van) / 2;
}

void Xuat(string hoten, int toan, int van, float dtb)
{
    cout << "Ho ten: " << hoten << "\n";
    cout << "Toan: " << toan << "\n";
    cout << "Van: " << van << "\n";
    cout << "DTB: " << dtb;
}
```

Chương trình 2:

```
#include<iostream>
#include<string>
using namespace std;

struct HocSinh
{
    string hoten;
    int toan;
    int van;
    float dtb;
};
typedef struct HocSinh HOCSINH;

void Nhap(HOCSINH &);
void XuLy(HOCSINH &);
void Xuat(HOCSINH);

int main()
{
    HOCSINH hs;
    Nhap(hs);
    XuLy(hs);
    Xuat(hs);
    return 1;
}

void Nhap(HOCSINH &hs)
{
    cout << "Nhap ho ten: ";
    getline(cin, hs.hoten);
    cout << "Nhap toan: ";
    cin >> hs.toan;
    cout << "Nhap van: ";
    cin >> hs.van;
}

void XuLy(HOCSINH& hs)
{
    hs.dtb = (float)(hs.toan + hs.van) / 2;
}

void Xuat(HOCSINH hs)
{
    cout << "Ho ten: " << hs.hoten << "\n";
    cout << "Toan: " << hs.toan << "\n";
    cout << "Van: " << hs.van << "\n";
    cout << "DTB: " << hs.dtb;
}
```

Bài 1: Viết chương trình nhập vào một phân số. Hãy cho biết phân số đó là phân số âm hay dương hay bằng không.

```
#include<iostream>
using namespace std;

struct phanso
{
    float tu;
    float mau;
};
typedef struct phanso PHANSO;

void Nhap(PHANSO&);
void Xuat(PHANSO);
int KiemTra(PHANSO);

int main()
{
    PHANSO x;
    Nhap(x);
    int kq = KiemTra(x);
    Xuat(x);
    switch (kq)
    {
        case 1: cout << "\nPhan so > 0 ";
                break;
        case -1: cout << "\nPhan so < 0";
                break;
        case 0: cout << "\nPhan so = 0";
                break;
    }
    return 1;
}

void Nhap(PHANSO &x)
{
    cout << "Nhap tu so: ";
    cin >> x.tu;
    cout << "Nhap mau so: ";
    cin >> x.mau;
}

void Xuat(PHANSO x)
{
    cout << "Tu so = " << x.tu<<"\n";
    cout << "Mau so = " << x.mau;
}

int KiemTra(PHANSO x)
{
    if (x.tu * x.mau > 0)
        return 1;
    if (x.tu * x.mau < 0)
        return -1;
    return 0;
}
```

Bài 2: Viết chương trình nhập tọa độ hai điểm trong không gian. Tính khoảng cách giữa chúng và xuất kết quả.

```
#include<iostream>
using namespace std;

struct diem
{
    float x;
    float y;
};
typedef struct diem DIEM;

void Nhap(DIEM&);
void Xuat(DIEM);
float KhoangCach(DIEM, DIEM);

int main()
{
    DIEM P, Q;
    cout << "Nhap diem thu nhat: \n";
    Nhap(P);
    cout << "Nhap diem thu hai: \n";
    Nhap(Q);
    float kc = KhoangCach(P, Q);
    Xuat(P);
    Xuat(Q);
    cout << "Khoang cach giua hai diem = " << kc;
    return 1;
}

void Nhap(DIEM& P)
{
    cout << "Nhap x: ";
    cin >> P.x;
    cout << "Nhap y: ";
    cin >> P.y;
}

void Xuat(DIEM P)
{
    cout << "(" << P.x << ", " << P.y << ")\n";
}

float KhoangCach(DIEM P, DIEM Q)
{
    float kc = sqrt((P.x - Q.x) * (P.x - Q.x) + (P.y - Q.y) * (P.y - Q.y));
    return kc;
}
```

Bài 3: Viết chương trình nhập vào hai phân số. Tìm phân số lớn nhất và xuất kết quả.

```
#include<iostream>
using namespace std;

struct phanso
{
    float tu;
    float mau;
};
typedef struct phanso PHANSO;

void Nhap(PHANSO&);
void Xuat(PHANSO);
int KiemTra(PHANSO);
PHANSO TimPhanSoLonNhat(PHANSO, PHANSO);

int main()
{
    PHANSO x, y;
    Nhap(x);
    Nhap(y);
    PHANSO kq = TimPhanSoLonNhat(x, y);
    Xuat(x);
    Xuat(y);
    cout << "Phan so lon nhat la: ";
    Xuat(kq);
    return 1;
}

void Nhap(PHANSO &x)
{
    cout << "Nhap tu so: ";
    cin >> x.tu;
    cout << "Nhap mau so: ";
    cin >> x.mau;
}

void Xuat(PHANSO x)
{
    cout << x.tu << "/" << x.mau << "\n";
}

int KiemTra(PHANSO x)
{
    if (x.tu * x.mau > 0)
        return 1;
    if (x.tu * x.mau < 0)
        return -1;
    return 0;
}

PHANSO Hieu(PHANSO x, PHANSO y)
{
    PHANSO temp;
    temp.tu = x.tu * y.mau - y.tu * x.mau;
```

```

        temp.mau = x.mau * y.mau;
        return temp;
    }

    PHANSO TimPhanSoLonNhat(PHANSO x, PHANSO y)
    {
        PHANSO temp = Hieu(x, y);
        if (KiemTra(temp) == -1)
            return y;
        return x;
    }

```

Bài 4: Viết chương trình nhập vào hai phân số. Tính tổng, hiệu, tích, thương giữa chúng và xuất kết quả.

```

#include<iostream>
using namespace std;

struct phanso
{
    int tu;
    int mau;
};
typedef struct phanso PHANSO;

void Nhap(PHANSO&);
void Xuat(PHANSO);
void RutGon(PHANSO&);
PHANSO Tong(PHANSO, PHANSO);
PHANSO Hieu(PHANSO, PHANSO);
PHANSO Tich(PHANSO, PHANSO);
PHANSO Thuong(PHANSO, PHANSO);

int main()
{
    PHANSO ps1, ps2;
    cout << "Nhap phan so thu nhat: \n";
    Nhap(ps1);
    cout << "Nhap phan so thu hai: \n";
    Nhap(ps2);
    PHANSO tong = Tong(ps1, ps2);
    PHANSO hieu = Hieu(ps1, ps2);
    PHANSO tich = Tich(ps1, ps2);
    PHANSO thuong = Thuong(ps1, ps2);
    RutGon(tong);
    RutGon(hieu);
    RutGon(tich);
    RutGon(thuong);
    Xuat(ps1);
    Xuat(ps2);
    cout << "Tong = ";
    Xuat(tong);
    cout << "Hieu = ";
    Xuat(hieu);
    cout << "Tich = ";

```

```

        Xuat(tich);
        cout << "Thuong = ";
        Xuat(thuong);
        return 1;
    }

void Nhap(PHANSO& ps)
{
    cout << "Nhap tu so: ";
    cin >> ps.tu;
    cout << "Nhap mau so: ";
    cin >> ps.mau;
}

void Xuat(PHANSO ps)
{
    cout << ps.tu << "/" << ps.mau << "\n";
}

void RutGon(PHANSO &ps)
{
    int a = abs(ps.tu);
    int b = abs(ps.mau);
    while (a * b != 0)
    {
        if (a > b)
            a = a - b;
        else
            b = b - a;
    }
    ps.tu = ps.tu / (a + b);
    ps.mau = ps.mau / (a + b);
}

PHANSO Tong(PHANSO ps1, PHANSO ps2)
{
    PHANSO temp;
    temp.tu = ps1.tu * ps2.mau + ps2.tu * ps1.mau;
    temp.mau = ps1.mau * ps2.mau;
    return temp;
}

PHANSO Hieu(PHANSO ps1, PHANSO ps2)
{
    PHANSO temp;
    temp.tu = ps1.tu * ps2.mau - ps2.tu * ps1.mau;
    temp.mau = ps1.mau * ps2.mau;
    return temp;
}

PHANSO Tich(PHANSO ps1, PHANSO ps2)
{
    PHANSO temp;
    temp.tu = ps1.tu * ps2.tu;
    temp.mau = ps1.mau * ps2.mau;
    return temp;
}

```

```

PHANSO Thuong(PHANSO ps1, PHANSO ps2)
{
    PHANSO temp;
    temp.tu = ps1.tu * ps2.mau;
    temp.mau = ps1.mau * ps2.tu;
    return temp;
}

```

Bài 5: Viết chương trình nhập vào 2 số phức. Tính tổng, hiệu, tích và xuất kết quả.

```

#include<iostream>
using namespace std;

struct sophuc
{
    float thuc;
    float ao;
};
typedef struct sophuc SOPHUC;

void Nhap(SOPHUC&);
void Xuat(SOPHUC);
SOPHUC Tong(SOPHUC, SOPHUC);
SOPHUC Hieu(SOPHUC, SOPHUC);
SOPHUC Tich(SOPHUC, SOPHUC);
SOPHUC Thuong(SOPHUC, SOPHUC);

int main()
{
    SOPHUC x, y;
    Nhap(x);
    Nhap(y);
    SOPHUC tong = Tong(x, y);
    SOPHUC hieu = Hieu(x, y);
    SOPHUC tich = Tich(x, y);
    SOPHUC thuong = Thuong(x, y);
    Xuat(x);
    Xuat(y);
    cout << "Tong = ";
    Xuat(tong);
    cout << "Hieu = ";
    Xuat(hieu);
    cout << "Tich = ";
    Xuat(tich);
    cout << "Thuong = ";
    Xuat(thuong);
    return 1;
}

void Nhap(SOPHUC& x)
{
    cout << "Nhap phan thuc: ";
    cin >> x.thuc;
    cout << "Nhap phan ao: ";
    cin >> x.ao;
}

```



```

void Xuat(SOPHUC x)
{
    cout << x.thuc << " + " << x.ao << "i\n";
}

SOPHUC Tong(SOPHUC x, SOPHUC y)
{
    SOPHUC temp;
    temp.thuc = x.thuc + y.thuc;
    temp.ao = x.ao + y.ao;
    return temp;
}

SOPHUC Hieu(SOPHUC x, SOPHUC y)
{
    SOPHUC temp;
    temp.thuc = x.thuc - y.thuc;
    temp.ao = x.ao - y.ao;
    return temp;
}

SOPHUC Tich(SOPHUC x, SOPHUC y)
{
    SOPHUC temp;
    temp.thuc = x.thuc * y.thuc - x.ao * y.ao;
    temp.ao = x.thuc * y.ao + x.ao * y.thuc;
    return temp;
}

SOPHUC Thuong(SOPHUC x, SOPHUC y)
{
    SOPHUC temp;
    temp.thuc = (x.thuc * y.thuc + x.ao * y.ao) / (y.thuc * y.thuc + y.ao * y.ao);
    temp.ao = (x.ao * y.thuc - x.thuc * y.ao) / (y.thuc * y.thuc + y.ao * y.ao);
    return temp;
}

```

Bài 6: Viết chương trình nhập vào một ngày. Tìm ngày kế tiếp và xuất kết quả.

```

#include <iostream>
using namespace std;

struct ngay
{
    int ng;
    int th;
    int nm;
};
typedef struct ngay NGAY;

void Nhap(NGAY&);
void Xuat(NGAY);
int KiemTraNamNhuan(NGAY);
long SoThuTu(NGAY);
int STTTrongNam(NGAY);

```

```

NGAY TimNgay(long);
NGAY TimNgay(int, int);
NGAY TimNgayKeTiep(NGAY);

int main()
{
    NGAY x;
    Nhap(x);
    Xuat(x);
    NGAY kq = TimNgayKeTiep(x);
    cout << "Ngay ke tiep: ";
    Xuat(kq);
    return 1;
}

void Nhap(NGAY& x)
{
    cout << "Nhap ngay: ";
    cin >> x.ng;
    cout << "Nhap thang: ";
    cin >> x.th;
    cout << "Nhap nam: ";
    cin >> x.nm;
}

void Xuat(NGAY x)
{
    cout << x.ng << "/" << x.th << "/" << x.nm << "\n";
}

int KiemTraNamNhuan(NGAY x)
{
    if (x.nm % 4 == 0 && x.nm % 100 != 0)
        return 1;
    if (x.nm % 400 == 0)
        return 1;
    return 0;
}

long SoThuTu(NGAY x)
{
    int nam = 1;
    long stt = 0;
    for (int i = 1; i < x.nm; i++)
    {
        stt = stt + 365;
        NGAY temp = { 1, 1, i };
        if (KiemTraNamNhuan(temp) == 1)
            stt = stt + 1;
    }
    return stt + STTTrongNam(x);
}

int STTTrongNam(NGAY x)
{
    int ngaythang[12] = { 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 };
    if (KiemTraNamNhuan(x) == 1)
        ngaythang[1] = 29;
}

```

```

    int stt = 0;
    for (int i = 1; i < x.th; i++)
    {
        stt = stt + ngaythang[i - 1];
    }
    return stt + x.ng;
}

NGAY TimNgay(long stt)
{
    int nam = 1;
    int songay = 365;
    while (stt - songay > 0)
    {
        stt = stt - songay;
        nam++;
        NGAY temp = { 1, 1, nam };
        if (KiemTraNamNhuan(temp) == 1)
            songay = 366;
        else
            songay = 365;
    }
    return TimNgay(nam, stt);
}

NGAY TimNgay(int nam, int stt)
{
    int ngaythang[12] = { 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 };
    NGAY x = { 1, 1, nam };
    if (KiemTraNamNhuan(x) == 1)
        ngaythang[1] = 29;
    x.th = 1;
    while (stt - ngaythang[x.th - 1] > 0)
    {
        stt = stt - ngaythang[x.th - 1];
        x.th++;
    }
    x.ng = stt;
    return x;
}

NGAY TimNgayKeTiep(NGAY x)
{
    long stt = SoThuTu(x);
    stt++;
    return TimNgay(stt);
}

```

Bài 7: Viết chương trình nhập vào một ngày. Tìm ngày hôm qua và xuất kết quả.

```
#include <iostream>
using namespace std;

struct ngay
{
    int ng;
    int th;
    int nm;
};
typedef struct ngay NGAY;

void Nhap(NGAY&);
void Xuat(NGAY);
int KiemTraNamNhuan(NGAY);
long SoThuTu(NGAY);
int STTTrongNam(NGAY);
NGAY TimNgay(long);
NGAY TimNgay(int, int);
NGAY TimNgayHomQua(NGAY);

int main()
{
    NGAY x;
    Nhap(x);
    Xuat(x);
    NGAY kq = TimNgayHomQua(x);
    cout << "Ngày hôm qua: ";
    Xuat(kq);
    return 1;
}

void Nhap(NGAY& x)
{
    cout << "Nhập ngày: ";
    cin >> x.ng;
    cout << "Nhập tháng: ";
    cin >> x.th;
    cout << "Nhập năm: ";
    cin >> x.nm;
}

void Xuat(NGAY x){
    cout << x.ng << "/" << x.th << "/" << x.nm << "\n";
}

int KiemTraNamNhuan(NGAY x)
{
    if (x.nm % 4 == 0 && x.nm % 100 != 0)
        return 1;
    if (x.nm % 400 == 0)
        return 1;
    return 0;
}
```

```

long SoThuTu(NGAY x)
{
    int nam = 1;
    int songay = 365;
    long stt = 0;
    for (int i = 1; i < x.nm; i++)
    {
        stt = stt + 365;
        NGAY temp = { 1, 1, i };
        if (KiemTraNamNhuan(temp) == 1)
            stt = stt + 1;
    }
    return stt + STTTrongNam(x);
}

int STTTrongNam(NGAY x)
{
    int ngaythang[12] = { 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 };
    if (KiemTraNamNhuan(x) == 1)
        ngaythang[1] = 29;
    int stt = 0;
    for (int i = 1; i < x.th; i++)
    {
        stt = stt + ngaythang[i - 1];
    }
    return stt + x.ng;
}

NGAY TimNgay(long stt)
{
    int nam = 1;
    int songay = 365;
    while (stt - songay > 0)
    {
        stt = stt - songay;
        nam++;
        NGAY temp = { 1, 1, nam };
        if (KiemTraNamNhuan(temp) == 1)
            songay = 366;
        else
            songay = 365;
    }
    return TimNgay(nam, stt);
}

NGAY TimNgay(int nam, int stt)
{
    int ngaythang[12] = { 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 };
    NGAY x = { 1, 1, nam };
    if (KiemTraNamNhuan(x) == 1)
        ngaythang[1] = 29;
    x.th = 1;
    while (stt - ngaythang[x.th - 1] > 0)
    {
        stt = stt - ngaythang[x.th - 1];
        x.th++;
    }
    x.ng = stt;
}

```

```

        return x;
    }

    NGAY TimNgayHomQua(NGAY x)
    {
        long stt = SoThuTu(x);
        stt--;
        return TimNgay(stt);
    }

```

Bài 8: Viết chương trình nhập tọa độ 3 đỉnh A, B, C của 1 tam giác trong mặt phẳng Oxy. Tính chu vi, diện tích và tìm tọa độ trọng tâm.

```

#include<iostream>
using namespace std;

struct diem
{
    float x;
    float y;
};
typedef struct diem DIEM;

struct tamgiac
{
    DIEM A;
    DIEM B;
    DIEM C;
};
typedef struct tamgiac TAMGIAC;

void Nhap(DIEM &);
void Xuat(DIEM);
void Nhap(TAMGIAC &);
void Xuat(TAMGIAC);
float KhoangCach(DIEM, DIEM);
float ChuVi(TAMGIAC);
float DienTich(TAMGIAC);

int main()
{
    TAMGIAC t;
    Nhap(t);
    float P = ChuVi(t);
    float S = DienTich(t);
    Xuat(t);
    cout << "Chu vi = " << P << "\n";
    cout << "Dien tich = " << S;
    return 1;
}

void Nhap(DIEM& P)
{
    cout << "Nhap x: ";
    cin >> P.x;
    cout << "Nhap y: ";
    cin >> P.y;
}

```

```

void Xuat(DIEM P)
{
    cout << "(" << P.x << ", " << P.y << ")\n";
}

float KhoangCach(DIEM P, DIEM Q)
{
    float kc = sqrt((P.x - Q.x) * (P.x - Q.x) + (P.y - Q.y) * (P.y - Q.y));
    return kc;
}

void Nhap(TAMGIAC &t)
{
    cout << "Nhap diem A : \n";
    Nhap(t.A);
    cout << "Nhap diem B : \n";
    Nhap(t.B);
    cout << "Nhap diem C : \n";
    Nhap(t.C);
}

void Xuat(TAMGIAC t)
{
    cout << "Diem A: ";
    Xuat(t.A);
    cout << "Diem B: ";
    Xuat(t.B);
    cout << "Diem C: ";
    Xuat(t.C);
}

float ChuVi(TAMGIAC t)
{
    float a = KhoangCach(t.B, t.C);
    float b = KhoangCach(t.A, t.C);
    float c = KhoangCach(t.A, t.B);
    return (a + b + c);
}

float DienTich(TAMGIAC t)
{
    float a = KhoangCach(t.B, t.C);
    float b = KhoangCach(t.A, t.C);
    float c = KhoangCach(t.A, t.B);
    float p = (a + b + c)/2;
    return sqrt(p * (p - a) * (p - b) * (p - c));
}

```

Bài 9: Viết chương trình nhập tọa độ tâm và bán kính của một đường tròn. Tính diện tích và chu vi đường tròn

```

#include<iostream>
using namespace std;

struct diem
{
    float x;

```

```

        float y;
    };
    typedef struct diem DIEM;

    struct duongtron
    {
        DIEM O;
        float R;
    };
    typedef struct duongtron DUONGTRON;

    void Nhap(DIEM&);
    void Xuat(DIEM);
    void Nhap(DUONGTRON&);
    void Xuat(DUONGTRON);
    float ChuVi(DUONGTRON);
    float DienTich(DUONGTRON);

    int main()
    {
        DUONGTRON c;
        Nhap(c);
        float P = ChuVi(c);
        float S = DienTich(c);
        Xuat(c);
        cout << "\nChu vi = " << P;
        cout << "\nDien tich = " << S;
        return 1;
    }

    void Nhap(DIEM& P)
    {
        cout << "Nhap x: ";
        cin >> P.x;
        cout << "Nhap y: ";
        cin >> P.y;
    }

    void Xuat(DIEM P)
    {
        cout << "(" << P.x << ", " << P.y << ")\n";
    }

    void Nhap(DUONGTRON &c)
    {
        cout << "Nhap tam O: \n";
        Nhap(c.O);
        cout << "Nhap ban kinh R: \n";
        cin >> c.R;
    }

    void Xuat(DUONGTRON c)
    {
        cout << "Tam O: ";
        Xuat(c.O);
        cout << "Ban kinh R: " << c.R;
    }

```



```
float ChuVi(DUONGTRON c)
{
    return 2 * 3.14 * c.R;
}

float DienTich(DUONGTRON c)
{
    return 3.14 * c.R * c.R;
}
```