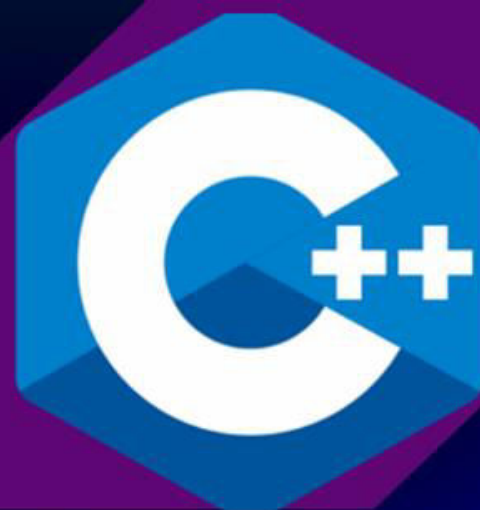


http://

Tuhoc.cc



LẬP TRÌNH



Dễ hiểu

LESSON

25.1

Kiểu cấu trúc C++

Khai báo cấu trúc -struct C++



@galailaptrinh



1

Khai báo cấu trúc C++

□ 1. Tổng quát :

- ✓ Ở các bài trước chúng ta đã được học về các kiểu dữ liệu cơ bản trong C++:
char, int, long, float, double....
- ✓ Tuy nhiên trong thực tế đôi khi cần sử dụng các kiểu dữ liệu phức tạp hơn rất nhiều → Sử dụng *struct* : là kiểu dữ liệu do ta tự định nghĩa
- ✓ *Struct*: có thể hiểu đơn giản là 1 tập các biến được khai báo sau keyword *struct*.



1

Khai báo cấu trúc C++

2. Cú pháp chung:

struct *tên_cấu_trúc* {

// biến thành viên, hay còn gọi là thuộc tính

kiểu_dữ_liệu tên_biến_thành_viên_1;

kiểu_dữ_liệu tên_biến_thành_viên_2;

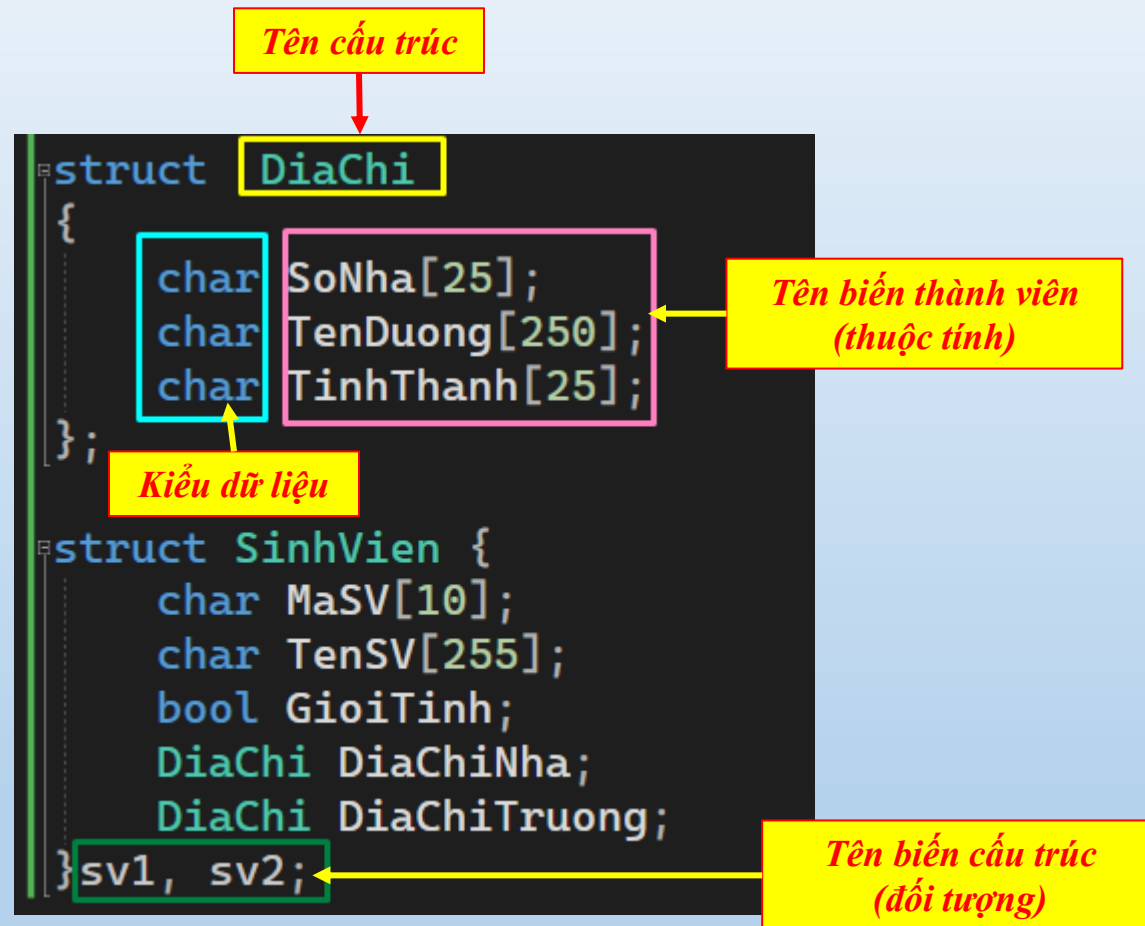
kiểu_dữ_liệu tên_biến_thành_viên_3;

.....

} *varName1, varName2 ;*

// tên các biến cấu trúc- các đối tượng

- ✓ Trong struct có thể khai báo nhiều biến thành viên
- ✓ Trong struct có thể chứa các struct khác



```
struct DiaChi
{
    char SoNha[25];
    char TenDuong[250];
    char TinhThanh[25];
};

struct SinhVien {
    char MaSV[10];
    char TenSV[255];
    bool GioiTinh;
    DiaChi DiaChiNha;
    DiaChi DiaChiTruong;
} sv1, sv2;
```



http://

Tuhoc.cc



LẬP TRÌNH



Dễ hiểu

LESSON

25.2

Kiểu cấu trúc C++

tạo đối tượng struct C++



@galailaptrinh



2

Tạo đối tượng struct C++

❑ 3. Có 2 cách tạo đối tượng :

✓ *Cách 1 khởi tạo ngay kèm lúc tạo struct*

✓ *Cách 2: Tạo như khai báo biến mới(có thể hiểu struct như 1 kiểu dữ liệu mới)*

```
struct SinhVien {
    char MaSV[10];
    char TenSV[255];
    bool GioiTinh;
    DiaChi DiaChiNha;
    DiaChi DiaChiTruong;
} sv1, sv2;
```

```
int main()
{
    //tạo các đối tượng của struct cách 2
    SinhVien sv3;
    SinhVien sv4;
}
```



3

Gán giá trị cho biến thành viên C++

//3.1 Cách 1

```
//{Tên_biến_cấu_trúc} = { {Giá_trị_1}, {Giá_trị_2}, ..., {Giá_trị_n} };
sv1 = { {"02"}, {"Ly thong"}, {true}, {"100"}, {"Nguyen Chi Thanh"}, {"Ha Noi"}} };
```

//3.2 Gán cách 2:

```
//có thể cần đối với 1 số trình biên dịch
#include "cstring"
strcpy_s(sv2.MaSV, sizeof(sv2.MaSV), "01");
strcpy_s(sv2.TenSV, sizeof(sv2.TenSV), "Ly Vac Dau");
sv2.GioiTinh = false;
```



4

Truy cập thuộc tính struct C++

❑ 4. Truy cập các biến thành viên (thuộc tính) :

Sử dụng *tên_đối_tượng.biến_thành_viên*

```
//Truy cập xuất thông tin sv1
cout << "Thông tin sv1 là: " << endl;
cout << "Mã SV: " << sv1.MaSV << endl;
cout << "Họ và Tên: " << sv1.TenSV << endl;
cout << "Giới tính: " << (sv1.GioiTinh == true ? "Nam" : "Nu") << endl;
cout << "Số nhà sv1 : " << sv1.DiaChiNha.SoNha << endl;
cout << "tên đường sv1 : " << sv1.DiaChiNha.TenDuong << endl;
cout << "tỉnh thành sv1: " << sv1.DiaChiNha.TinhThanh << endl;

//xuất thông tin sv2
cout << "Thông tin sv2 là: " << endl;
cout << "Mã SV: " << sv2.MaSV << endl;
cout << "Họ và Tên: " << sv2.TenSV << endl;
cout << "Giới tính: " << (sv2.GioiTinh == true ? "Nam" : "Nu") << endl;
```



4

Truy cập thuộc tính struct C++

❑ 4. Truy cập các biến thành viên (thuộc tính) :

Sử dụng tên_đối_tượng.biến_thành_viên

```
Thông tin sv1 là:  
Ma SV: 02  
Ho và Ten: Ly thong  
Giới tính: Nam  
Số nhà sv1 : 100  
tên dương sv1 : Nguyen Chi Thanh  
tỉnh thành sv1: Hà Nội  
Thông tin sv2 là:  
Ma SV: 01  
Ho và Ten: Ly Vac Dau  
Giới tính: Nữ  
Địa chỉ nhà của sv2 :  
số nhà 45 Tên dương hoàng quốc việt Tỉnh thành hà nội
```

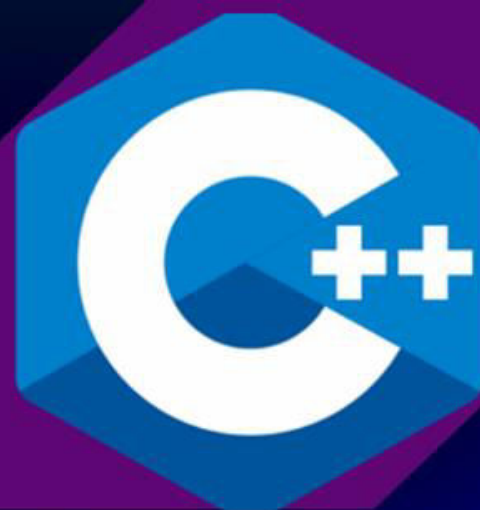


http://

Tuhoc.cc



LẬP TRÌNH



Dễ hiểu

LESSON

25.3

Kiểu cấu trúc C++

Sao chép giá trị struct



@galailaptrinh



5

Gán - Sao chép giá trị

```
#include "iostream"
using namespace std;

struct Book
{
    char tieuDe[100];
    char tacGia[100];
    float giaSach;
};
```

```
Moi cu nhap ten sach: lap trinh co ban
Moi cu nhap tac gia: ga lai xap xinh
Moi cu nhap gia sach: 99999
Thong tin quyen 2:
Ten sach: lap trinh co ban
tac gia: ga lai xap xinh
gia sach: 99999
Moi cu nhap gia thay doi quyen 1: 15000
gia sach quyen 1: 15000
gia sach quyen 2: 99999
dia chi o nho quyen 1: 00000050054FF820
dia chi o nho quyen 2: 00000050054FF910
```

```
int main()
{
    Book quyen1;
    cout << "Moi cu nhap ten sach: ";
    gets_s(quyen1.tieuDe);
    cout << "Moi cu nhap tac gia: ";
    gets_s(quyen1.tacGia);
    cout << "Moi cu nhap gia sach: ";
    cin >> quyen1.giaSach;

    //sao chép thông tin đối tượng quyen1 cho quyen2
    Book quyen2 = quyen1;

    //xuất thông tin
    cout << "Thong tin quyen 2: " << endl;
    cout << "Ten sach: " << quyen2.tieuDe << endl;
    cout << "tac gia: " << quyen2.tacGia << endl;
    cout << "gia sach: " << quyen2.giaSach << endl;
    //thay doi gia sach quyen 1
    cout << "Moi cu nhap gia thay doi quyen 1: ";
    cin >> quyen1.giaSach;
    //xuat gia sach quyen 1 va quyen 2 de kiem tra
    cout << "gia sach quyen 1: " << quyen1.giaSach << endl;
    cout << "gia sach quyen 2: " << quyen2.giaSach << endl;

    //check địa chỉ ô nhớ quyen1 và quyen2
    cout << "dia chi o nho quyen 1: " << &quyen1 << endl;
    cout << "dia chi o nho quyen 2: " << &quyen2 << endl;
}
```



http://

Tuhoc.cc



LẬP TRÌNH



Dễ hiểu

LESSON

25.4

Kiểu cấu trúc C++

Mảng cấu trúc Struct



@galailaptrinh



6

Mảng cấu trúc C++

- ✓ *Coi cấu trúc như 1 kiểu dữ liệu do ta tạo ra*
- ✓ *Khai báo như khai báo mảng thông thường*
- ✓ *Các phần tử cũng có thể dùng toán tử **.Dot** để truy cập các thuộc tính(các biến thành viên)*

```
struct SinhVien
{
    int ma;
    char ten[255];
};
```

```
//Khai báo giống mảng thông thường
int mang1[15]; //mảng 1, các pt là kiểu nguyên, chứa tối đa 15 pt

const int total = 4;
//khai báo mảng cấu trúc
SinhVien ds[total];
```



6

Mảng cấu trúc C++

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

struct SinhVien
{
    int ma;
    char ten[255];
};

//hàm nhập mảng
void NhapMang(SinhVien ds[], int total) {
    for (int i = 0; i < total; i++) {
        cout << "Nhap thong tin sv thu " << i+1 << endl;
        cout << "Nhap ma: ";
        cin >> ds[i].ma;
        //xóa bộ nhớ đệm cin
        cin.ignore();
        cout << "Nhap ten: ";
        gets_s(ds[i].ten, sizeof(ds[i].ten));
        cin.ignore();
    }
}

//hàm xuất mảng
void XuatMang(SinhVien ds[], int total) {
    for (int i = 0; i < total; i++) {
        cout << ds[i].ma << "\t" << ds[i].ten << endl;
    }
}
```

```
int main()
{
    //khải báo mảng với kiểu dữ liệu cơ bản C++
    int mang1[15];
    //khải báo với cấu trúc
    const int total = 4;
    SinhVien ds[total];

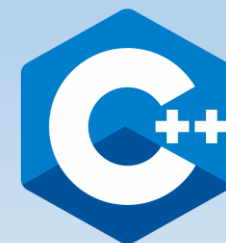
    NhapMang(ds, total);
    XuatMang(ds, total);
}
```

Nhap thong tin sv thu 2
 Nhap ma: 2
 Nhap ten: trinh thi dua leo

Nhap thong tin sv thu 3
 Nhap ma: 3
 Nhap ten: mai van dot

Nhap thong tin sv thu 4
 Nhap ma: 4
 Nhap ten: ngheo mai ngheo

1	hua van leo
2	trinh thi dua leo
3	mai van dot
4	ngheo mai ngheo

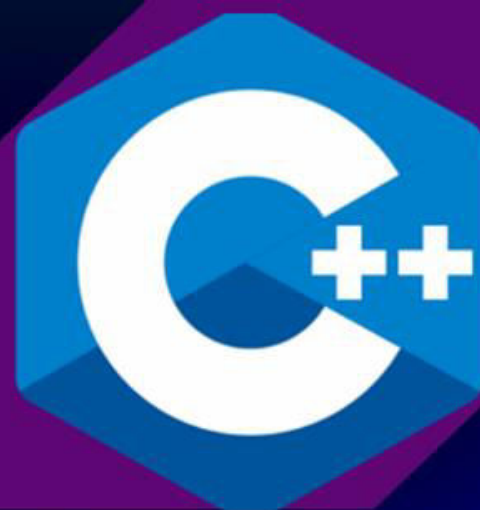


http://

Tuhoc.cc



LẬP TRÌNH



Dễ hiểu

LESSON

25.5

Kiểu cấu trúc C++

Con trỏ cấu trúc - Struct



@galailaptrinh



6

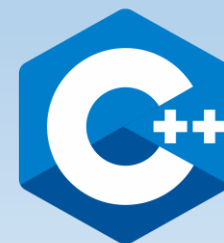
Con trỏ cấu trúc C++

❑ **6.1** C++ coi **struct** như 1 kiểu dữ liệu (kiểu dữ liệu này do lập trình viên định nghĩa) → **Cho phép sử dụng con trỏ như với biến thông thường**

```
struct SinhVien
{
    int ma;
    char ten[300];
};
int main()
{
    // Ví dụ con trỏ với biến thông thường
    //int* conTroNguyen = new int; // vừa khai báo, vừa cấp phát
    int* conTroNguyen; // khai báo
    conTroNguyen = new int; //cấp phát

    //2.1 khởi tạo đối tượng thông thường
    SinhVien sv1;
    //gán giá trị cho đối tượng sv1
    sv1 = { 1, "tran van nhong" };
    //xuất thông tin (dùng dot )
    cout << "ma sv1 là: " << sv1.ma << endl;
    cout << "ten sv1 là: " << sv1.ten << endl;
}
```

```
//khởi tạo đối tượng dùng con trỏ
SinhVien* psv2; //khai báo con trỏ
psv2 = new SinhVien;
//gán giá trị
*psv2 = { 2, "tran thach cao" };
//xuất thông tin cho cấu trúc con trỏ
```



6

Con trỏ cấu trúc C++

❑ 6.2 Truy cập các biến thành viên của biến con trỏ :

Sử dụng -> thay cho . dot

```
//khởi tạo đối tượng dùng con trỏ
SinhVien* psv2; //khai báo con trỏ
psv2 = new SinhVien;
//gán giá trị
*psv2 = { 2, "tran thach cao" };
//xuất thông tin cho cấu trúc con trỏ

//2.2 Truy cập biến thành viên đối với con trỏ thì dùng -> thay cho dot
cout << "ma sv2 la: " << psv2->ma << endl;
cout << "ten sv2 la: " << psv2->ten << endl;
```

```
ma sv1 la: 1
ten sv1 la: tran van nhong
ma sv2 la: 2
ten sv2 la: tran thach cao
```



6

Con trỏ cấu trúc C++

❑ 6.3 Gán con trỏ vào địa chỉ ô nhớ của biến cấu trúc(đối tượng thường)

✓ *Quản lý cùng vùng nhớ, thay đổi 1 sẽ thay đổi cả 2*

```
//3.Gán con trỏ vào địa chỉ ô nhớ của biến cấu trúc( đối tượng thường)
SinhVien* psv3; //khai báo con trỏ
psv3 = &sv1; // trỏ con trỏ psv3 vào địa chỉ ô nhớ của sv1
// lúc này psv3 và sv1 quản lý cùng ô nhớ, thay đổi 1 sẽ thay đổi cả 2
//xuất thông tin biến con trỏ psv3
cout << "ma sv3 la: " << psv3->ma << endl;
cout << "ten sv3 la: " << psv3->ten << endl;

//test thay đổi biến con trỏ psv3 :
*psv3 = { 99, "nguyen tan vang" };
//xuất thông tin sv1 và psv3 sau đổi
cout << psv3->ma << "\t" << psv3->ten << endl;
cout << sv1.ma << "\t" << sv1.ten << endl;
```

```
ma sv3 la: 1
ten sv3 la: tran van nhong
99      nguyen tan vang
99      nguyen tan vang
```



2

Bài tập C++ 27 - 29

□ Bài tập C++ 27

*Viết chương trình sử dụng cấu trúc
struct DiaChi:*

tên đường

Quận

tỉnh thành

struct NhanVien:

Ma (int),

ten (char[])

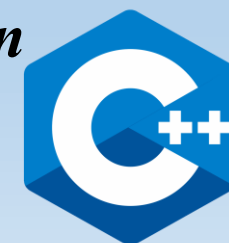
tuoi (int)

DiaChi CoQuan

DiaChia NhaRieng

1. Tạo 2 đối tượng là nv1, nv2 có đầy đủ toàn bộ thông tin trên, và xuất ra màn hình

2. Khai báo bằng con trỏ



2

Bài tập C++ 27 - 29

□ Bài tập C++ 28

Sử dụng struc, có 2 biến thành viên là: tọa độ x, y

Viết hàm:

- 1. tính khoảng cách giữa 2 điểm*
- 2. Nhập vào 1 mảng các điểm, tính tổng độ dài các điểm đó*

□ Bài tập C++ 29

Sử dụng struc, có 2 biến thành viên là: tử số, mẫu số

Viết hàm:

- 1. Cộng, trừ, nhân, chia, tối giản phân số*



http://

Tuhoc.cc



LẬP TRÌNH



Dễ hiểu

LESSON

25.6

Giải bài tập 27

Kiểu cấu trúc C++



@galailaptrinh



5

Giải Bài tập 27

```
#include "iostream"
using namespace std;

struct DiaChi
{
    char tenDuong[300];
    char quan[300];
    char tinhThanh[300];
};

struct NhanVien
{
    int ma;
    char ten[300];
    int tuoi;
    DiaChi coQuan;
    DiaChi nhaRieng;
};
```

```
//tạo 2 đối tượng nv1 và nv2, khai báo thường
NhanVien nv1;
NhanVien nv2;
//gán thông tin
nv1 = { 1,"Do Tran Chua Chin",21,{"Tran Duy Hung","Hai Ba Tung","Ha Noi"},{ "Tran Hung Dao","Hoang Mai","Kon Tom" } };
nv2 = { 2,"Do Tran Nhu",49,{ "Lac Long Quan","Quan Thu Duc","Tp HCM" },{ "Au Duong Phong","Quan 3","Kon Lon" } };
//xuất thông tin
cout << "Thông tin nv1: " << endl;
cout << "Ma: " << nv1.ma << endl;
cout << "Ten: " << nv1.ten << endl;
cout << "Tuoi: " << nv1.tuoi << endl;
cout << "Co quan: " << nv1.coQuan.tenDuong << ", "
    << nv1.coQuan.quan << ", "
    << nv1.coQuan.tinhThanh << ", "
    << endl;
cout << "Nha Rieng: " << nv1.nhaRieng.tenDuong << ", "
    << nv1.nhaRieng.quan << ", "
    << nv1.nhaRieng.tinhThanh << endl;
```



5

Giải Bài tập 27

```
//nâng cao: khai báo bằng con trỏ
NhanVien* pnv3 = new NhanVien; // khai báo
//pnv3 = new NhanVien; //cấp phát bộ nhớ
//gán thông tin, lưu ý nếu struct chồng struc con thì gán bằng toán tử * có thể bị lỗi

*pnv3 = { 3,"Do Tran Nhu Nhung",21,{ "Truong Trinh","Quan 8","tp HCM" },{ "Tran Nhan Tong","Thanh Xuan","Ha Noi" } };
//gán thông thường dùng strcpy
/*pnv3->ma = 3;
strcpy_s(pnv3->ten, sizeof(pnv3->ten), "Do Tran Nhu Nhung");
pnv3->tuoi = 21;
strcpy_s(pnv3->coQuan.tenDuong, sizeof(pnv3->coQuan.tenDuong), "Truong Trinh");
strcpy_s(pnv3->coQuan.quan, sizeof(pnv3->coQuan.quan), "Quan 8");
strcpy_s(pnv3->coQuan.tinhThanh, sizeof(pnv3->coQuan.tinhThanh), "tp HCM");
strcpy_s(pnv3->nhaRieng.tenDuong, sizeof(pnv3->nhaRieng.tenDuong), "Tran Nhan Tong");
strcpy_s(pnv3->nhaRieng.quan, sizeof(pnv3->nhaRieng.quan), "Thanh Xuan");
strcpy_s(pnv3->nhaRieng.tinhThanh, sizeof(pnv3->nhaRieng.tinhThanh), "Ha Noi");*/

//xuất thông tin
cout << "Thong tin nv3: " << endl;
cout << "Ma: " << pnv3->ma << endl;
cout << "Ten: " << pnv3->ten << endl;
cout << "Tuoi: " << pnv3->tuoi << endl;
cout << "Co quan: " << pnv3->coQuan.tenDuong << ", "
    << pnv3->coQuan.quan << ", "
    << pnv3->coQuan.tinhThanh << ", "
    << endl;
cout << "Nha Rieng: " << pnv3->nhaRieng.tenDuong << ", "
    << pnv3->nhaRieng.quan << ", "
    << pnv3->nhaRieng.tinhThanh << endl;
```



http://

Tuhoc.cc



LẬP TRÌNH



Dễ hiểu

LESSON

25.7

Giải bài tập 28

Kiểu cấu trúc C++



@galailaptrinh



5

Giải Bài tập 28

```

/*
bt28 : Sử dụng struc, có 2 biến thành viên là: tọa độ x, y
Viết hàm:
1. tính khoảng cách giữa 2 điểm
2. Nhập vào 1 mảng các điểm, tính tổng độ dài các điểm đó
*/

#include "iostream"
using namespace std;

struct Diem
{
    double x;
    double y;
};

//hàm khoảng cách 2 điểm
double KhoangCach(Diem a, Diem b) {
    return sqrt(pow(a.x - b.x, 2) + pow(a.y - b.y, 2));
}

//2. Độ dài các điểm
double DoDaiCacDiem(Diem mang[], int soDiem) {
    double tongkc = 0;
    //chú ý: chạy đến soDiem-1 vì điểm cuối cùng kết thúc
    for (int i = 0; i < soDiem - 1; i++) {
        double kc = KhoangCach(mang[i], mang[i + 1]);
        tongkc += kc;
        //in thử để xem
        cout << "kc thu " << i << " = " << kc << endl;
    }
    return tongkc;
}

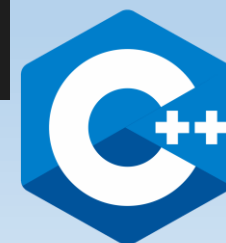
```

```

//hàm nhập tọa độ các điểm
void NhapDiem(Diem mang[], int soDiem) {
    for (int i = 0; i < soDiem; i++) {
        cout << "Nhap diem thu " << i + 1 << endl;
        cout << " Nhap toa do x: ";
        cin >> mang[i].x;
        cin.ignore(); //xóa bộ nhớ đệm
        cout << " Nhap toa do y: ";
        cin >> mang[i].y;
        cin.ignore(); //xóa bộ nhớ đệm
    }
}

//hàm xuất tọa độ các điểm đã nhập
void XuatDiem(Diem mang[], int soDiem) {
    for (int i = 0; i < soDiem; i++) {
        cout << mang[i].x << " - " << mang[i].y << ", ";
    }
}

```



5

Giải Bài tập 28

```
int main()
{
    //1. tính khoảng cách giữa 2 điểm
    //khai báo Diem a, Diem b;
    Diem a;
    Diem b;
    //gán giá trị
    a = { 1, 1 };
    b = { 3, 3 };
    //gọi hàm khoảng cách
    double kc = KhoangCach(a, b);
    cout << "Khoang cach 2 diem a,b :" << kc << endl;

    //2. Nhập danh sách điểm, tính tổng khoảng cách các điểm
    const int soDiem = 3;
    //khởi tạo mảng
    Diem mang[soDiem];
    NhapDiem(mang, soDiem);
    cout << "Danh sach toa do vua nhap : " << endl;
    XuatDiem(mang, soDiem);
    cout << endl;
    cout << "Tong do dai cac diem tren = " << DoDaiCacDiem(mang, soDiem);
}
```



http://

Tuhoc.cc



LẬP TRÌNH



Dễ hiểu

LESSON

25.8

Giải bài tập 29

Kiểu cấu trúc C++



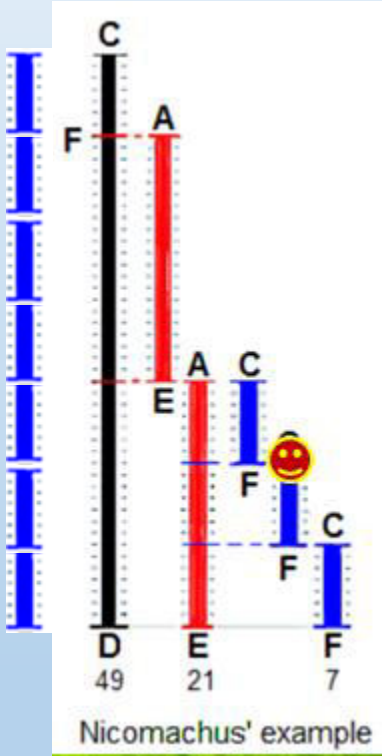
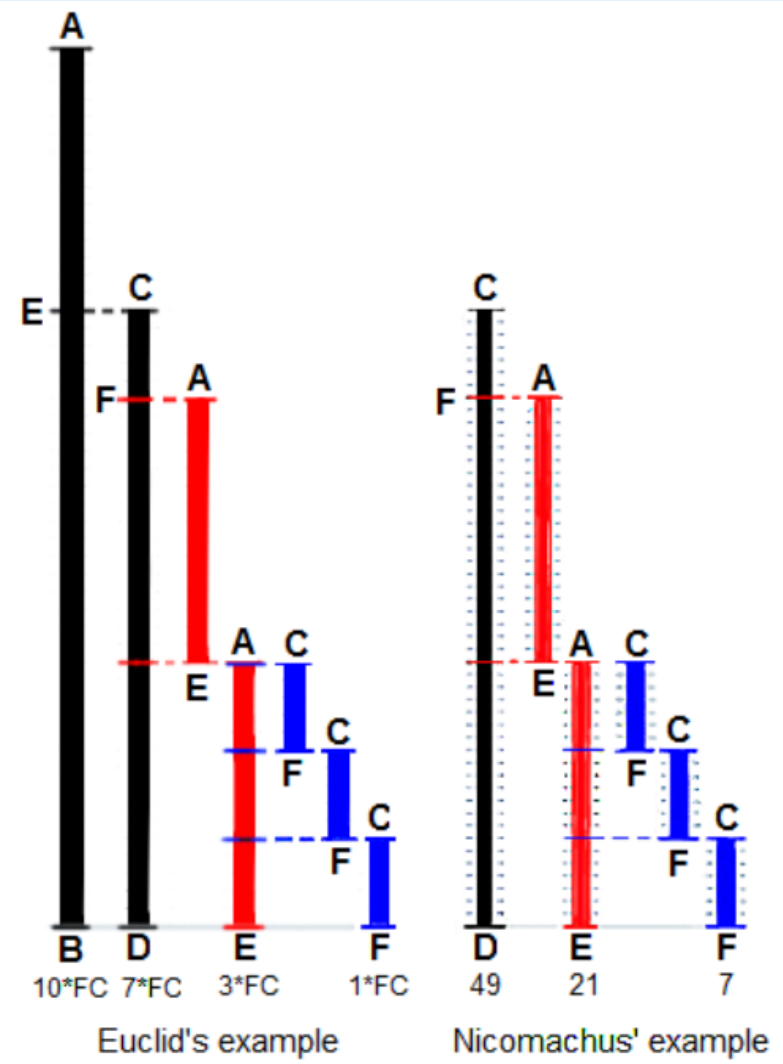
@galailaptrinh



5

Giải thuật euclid

Ví dụ 1: $\frac{7}{9} + \frac{3}{10} = \frac{70}{90} + \frac{27}{90} = \frac{97}{90}$.



5

Giải Bài tập 29

```
/*
bt29: cho phân số có cấu trúc: tử, mẫu
Viết hàm cộng, trừ, nhân, chia, tối giản phân số
*/

#include <iostream>
using namespace std;

struct PhanSo
{
    int tu, mau;
};

// hàm tính ước chung lớn nhất sử dụng tiên đề euclid để tìm
int UCLN(int a, int b)
{
    a = abs(a);
    b = abs(b);
    if (b == 0) return a;
    return UCLN(b, a % b);
}

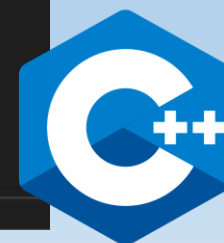
void ToiGianPhanSo(PhanSo& ps) // hàm tối giản phân số
{
    int ucln = UCLN(ps.tu, ps.mau);
    ps.tu /= ucln;
    ps.mau /= ucln;
}
```

```
PhanSo CongPhanSo(PhanSo ps1, PhanSo ps2) // hàm cộng phân số
{
    PhanSo kq;
    kq.tu = ps1.tu * ps2.mau + ps1.mau * ps2.tu;
    kq.mau = ps1.mau * ps2.mau;
    ToiGianPhanSo(kq);
    return kq;
}

PhanSo TruPhanSo(PhanSo ps1, PhanSo ps2) // hàm trừ phân số
{
    PhanSo kq;
    kq.tu = ps1.tu * ps2.mau - ps1.mau * ps2.tu;
    kq.mau = ps1.mau * ps2.mau;
    ToiGianPhanSo(kq);
    return kq;
}

PhanSo NhanPhanSo(PhanSo ps1, PhanSo ps2) // hàm nhân phân số
{
    PhanSo kq;
    kq.tu = ps1.tu * ps2.tu;
    kq.mau = ps1.mau * ps2.mau;
    ToiGianPhanSo(kq);
    return kq;
}

PhanSo ChiaPhanSo(PhanSo ps1, PhanSo ps2) // hàm chia phân số
{
    PhanSo kq;
    kq.tu = ps1.tu * ps2.mau;
    kq.mau = ps1.mau * ps2.tu;
    ToiGianPhanSo(kq);
    return kq;
}
```



5

Giải Bài tập 29

```
int main()
{
    PhanSo ps1, ps2, kq;
    cout << "Nhap phan so thu nhat: ";
    cin >> ps1.tu >> ps1.mau;
    cin.ignore();
    cout << "Nhap phan so thu hai: ";
    cin >> ps2.tu >> ps2.mau;
    cin.ignore();

    // Cộng hai phân số
    kq = CongPhanSo(ps1, ps2);
    cout << "Tong hai phan so: " << kq.tu << "/" << kq.mau << endl;

    // Trừ hai phân số
    kq = TruPhanSo(ps1, ps2);
    cout << "Hieu hai phan so: " << kq.tu << "/" << kq.mau << endl;

    // Nhân hai phân số
    kq = NhanPhanSo(ps1, ps2);
    cout << "Tich hai phan so: " << kq.tu << "/" << kq.mau << endl;

    // Chia hai phân số
    kq = ChiaPhanSo(ps1, ps2);
    cout << "Thuong hai phan so: " << kq.tu << "/" << kq.mau << endl;

    return 0;
}
```

```
Nhap phan so thu nhat: 1
2
Nhap phan so thu hai: 3
4
Tong hai phan so: 5/4
Hieu hai phan so: -1/4
Tich hai phan so: 3/8
Thuong hai phan so: 2/3
```

