



# Phương pháp lập trình hướng đối tượng

Tuần 02:

## Lớp và đối tượng



Phạm Tú San  
[ptsan@fit.hcmus.edu.vn](mailto:ptsan@fit.hcmus.edu.vn)

# Đặt vấn đề

---

- Viết chương trình nhập họ tên, điểm toán, điểm văn của một học sinh. Tính điểm trung bình và xuất kết quả



# Lập trình hướng thủ tục

```
struct HocSinh
{
    int MaHS;
    char HoTen[50];
    float DiemToan;
    float DiemVan;
};
```

Thành phần dữ liệu

```
void NhapHS(HocSinh &hs);
void XuatHS(HocSinh hs);
float DiemTB(HocSinh hs);
```

Thành phần xử lý

```
void main()
{
    HocSinh a;
    NhapHS(a);
    XuatHS(a);
}
```

Lấy hành  
động làm  
trung tâm

# Vấn đề phát sinh

- Giả sử một đoạn trong chương trình:

```
void main()  
{  
    HocSinh a;  
    NhapHS(a);  
    .....  
    a.DiemToan = -5;  
}
```

Sai!

- Kiểm soát dữ liệu thế nào?

Đóng gói dữ  
liệu!

# Đặt vấn đề

---

- Lập trình hướng đối tượng
  - Đóng gói dữ liệu, lấy dữ liệu làm trung tâm
  - Dữ liệu của ai do người đó quản lý
    - Muốn đổi điểm học sinh -> phải nhờ đối tượng học sinh làm giúp



# Lập trình hướng đối tượng

```
class CHocSinh
{
private:
    char mMSSV;
    char mHoTen[50];
    float mDiemToan;
    float mDiemVan;
public:
    void GanDiemToan(int diem)
    {
        if(diem>=0 && diem<=10)
            this->mDiemToan = diem;
    }
};
```

```
void main
{
    CHocSinh a;
a.mDiemToan = 5;

    a.GanDiemToan(-5);
}
```



# Lập trình hướng đối tượng

---

- **Object Oriented Programming (OOP)**
- Là **một cách tiếp cận** (hay phương pháp/kiểu) lập trình (design philosophy)
  - Các ngôn ngữ lập trình hướng đối tượng: C++, C#, Java..
- Chương trình hướng đối tượng là chương trình dựa trên sự tương tác giữa các **đối tượng** (**objects**)

**ĐỐI TƯỢNG** là trung tâm

# Đối tượng (object)

---

- **Đối tượng** – một thực thể cụ thể
  - thuộc tính - **attribute** (dữ liệu)
  - hành động - **method** (hoạt động, ứng xử, xử lý)
- Học sinh Nguyễn Văn, phân số  $p = \frac{1}{2}$ ...
- Trong thực tế
  - Thuộc tính – các biến, tham số hay hằng nội tại của một đối tượng – mô tả tính chất
  - Phương thức – hàm – phương tiện sử dụng





# Ví dụ đối tượng

## ● Ví dụ: Chương trình “Điểm trung bình 3 HS”

```
class HocSinh{
private:
    char hoten[31];
    int toan;
    int van;
public:
    HocSinh(char*
iHoTen);
    void diThi();
    void xuat();
    float tinhDTB();
};
```

```
void main() {
    HocSinh a("Việt"),
    b("Cường"),
    c("Thuần");
    a.diThi(); b.diThi();
    c.diThi();
    a.tinhDTB();
    b.tinhDTB();
    c.tinhDTB();
    a.xuat(); b.xuat();
    c.xuat();
}
```

Liệt kê một số đối tượng?  
Thuộc tính?  
Phương thức?

# Lớp đối tượng (Class)

---

- Đối tượng được trừu tượng hóa thành **lớp đối tượng - class**
- **Lớp đối tượng** là kiểu dữ liệu các đối tượng cùng loại
  - Jessica Alba, Bill Gates -> đối tượng cùng loại NGƯỜI
  - Lớp 11TH, lớp 12TH -> đối tượng cùng loại LỚP HỌC

```
public class Student  
{ }
```

```
Student objectStudent  
= new Student();
```

# Ví dụ class

---

```
class XeHoi
{
private:
    float mXangHienCo;
    float mDungTichBinhXang;
public:
    void Chay(float km);
    void DoXang(float xang);
}
```

# Các tính chất của class

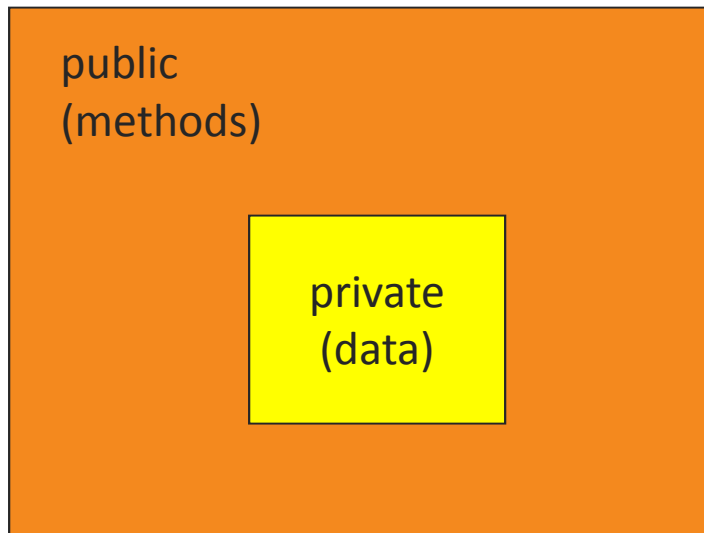
---

- **Đóng gói:** chứa đựng dữ liệu và các hàm/thủ tục liên quan
- **Che giấu dữ liệu:** các thực thể phần mềm khác không can thiệp trực tiếp vào dữ liệu bên trong được mà phải thông qua các phương thức cho phép



# Một cách thể hiện điển hình

- Che giấu dữ liệu và các “giải thuật” cụ thể ở bên trong lớp (class)



lớp đối tượng

```
class <Tên lớp>
{
    private:
        <Các thành phần riêng>
    public:
        <Các thành phần chung/công cộng>
};
```

# Phạm vi hoạt động (scope)

---

- **private**: chỉ sử dụng được khi đang ở bên trong lớp (class) đó

(Một cách giải thích khác: chỉ các hàm hay phương thức ở bên trong của lớp đó mới *nhìn thấy* được)

- **public**: có thể sử dụng ở cả bên trong và bên ngoài của lớp





# Ví dụ

```
class CHocSinh
{
private:
    char mHoTen[50];
    float mDiemToan;
    float mDiemVan;
public:
    void GanDiemToan(int diem)
    {
        if(diem>=0 && diem<=10)
            this->mDiemToan =
diem;
    }
};
```

```
void main
{
    CHocSinh a;
a.mDiemToan = -5;
    a.GanDiemToan(-5);
}
```

# Các tính chất của lập trình hướng đối tượng

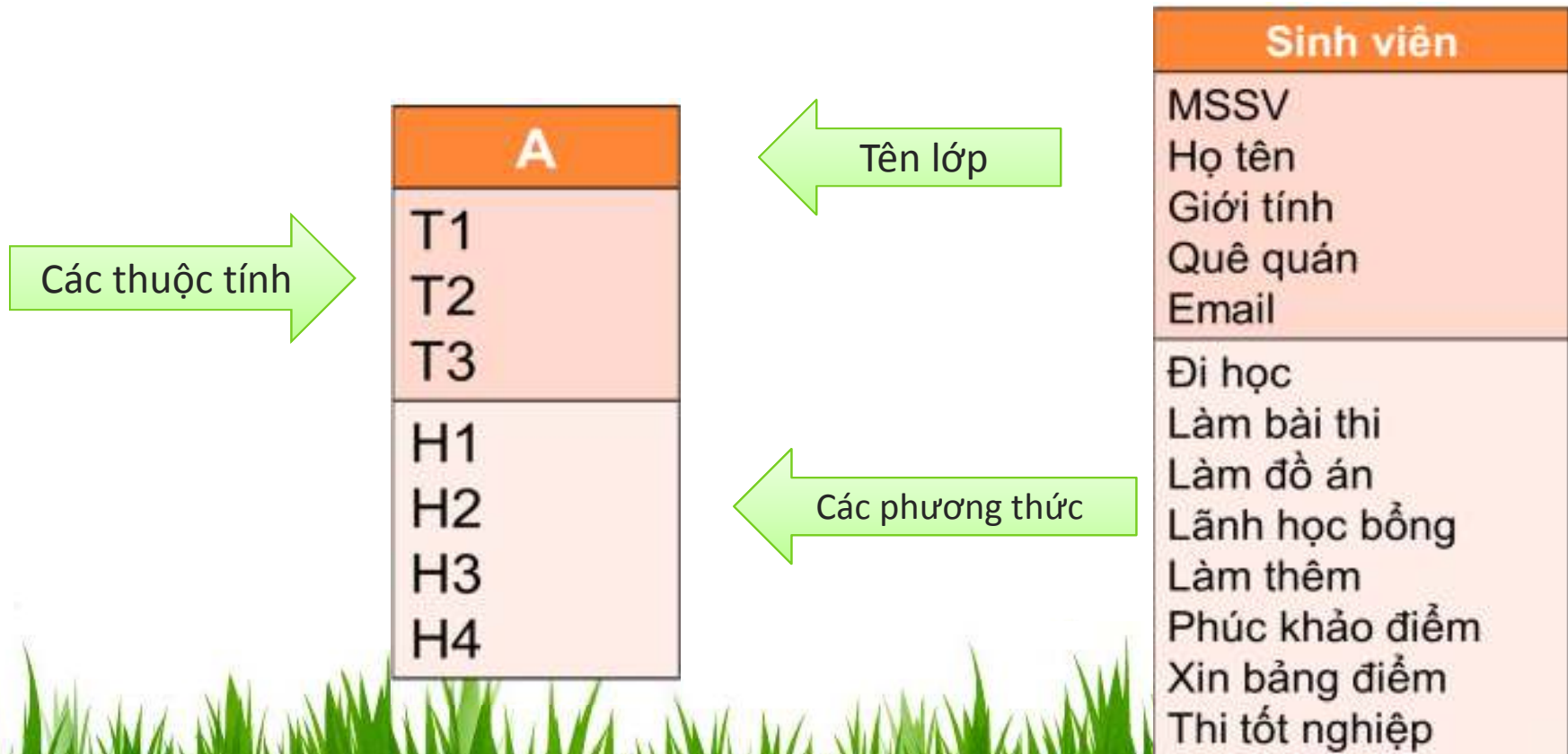
---

- 1 – tính trừu tượng (abstraction)
- 2 – tính đóng gói (encapsulation) và che giấu thông tin (information hiding)
- 3 – tính đa hình (polymorphism)
- 4 – tính kế thừa (inheritance)

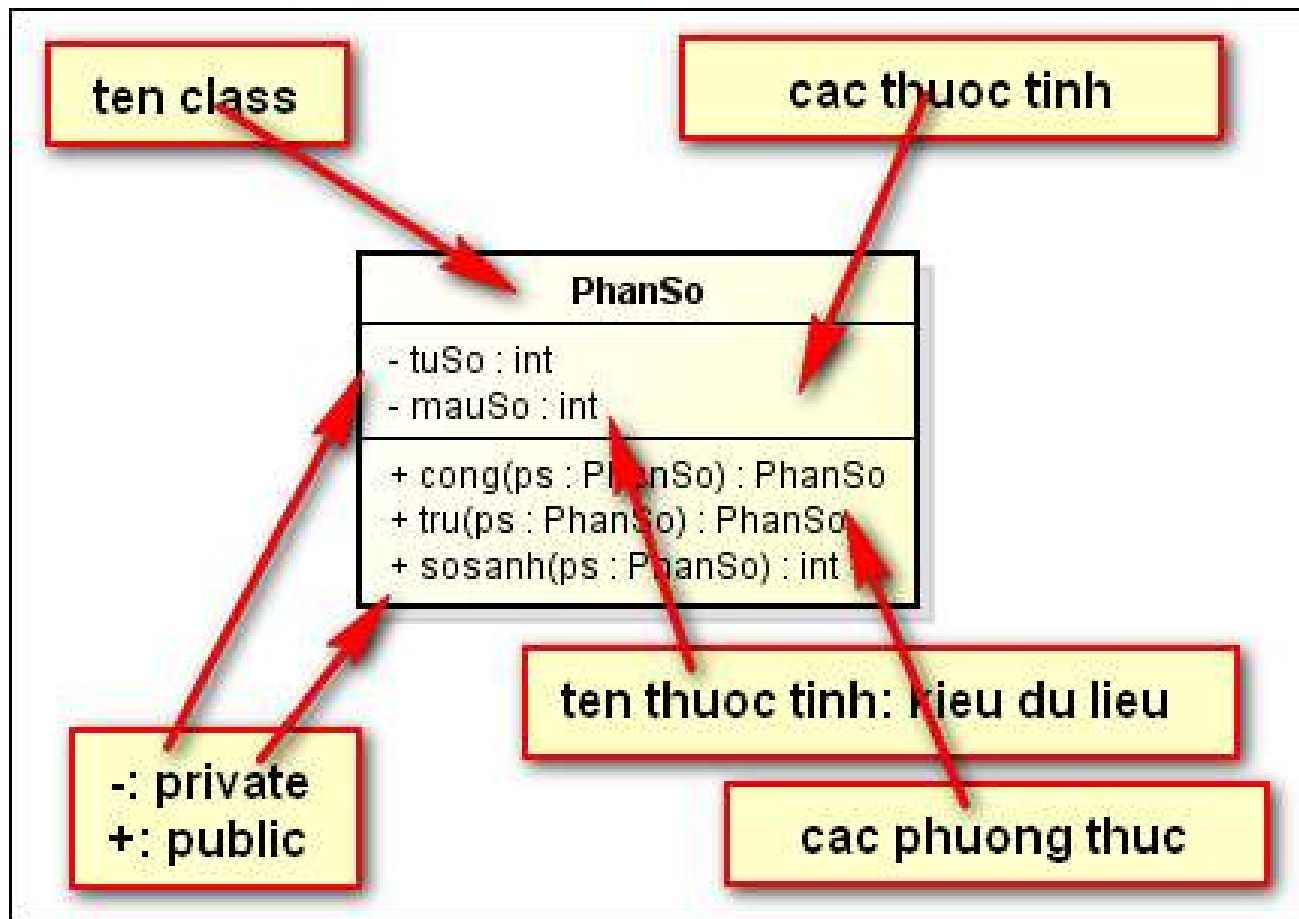


# Sơ đồ lớp – class diagram

- Lớp đối tượng A có 3 thuộc tính là T1, T2, T3 và 4 hành động là H1, H2, H3, H



# Sơ đồ lớp – mức cài đặt được



# Chương trình vẽ uml

---

- Một số chương trình hỗ trợ thiết kế sơ đồ UML
  - Astah
  - StarUML
  - Rational rose



## Toán tử phạm vi ::

---

- Dùng để chỉ dẫn thành phần thuộc tính hay phương thức thuộc một lớp đối tượng cụ thể nào đó
  - CHocSinh::GanDiemToan(int diem)





# Tách biệt khai báo và định nghĩa

```
//file HocSinh.h
class CHocSinh
{
    private:
        ...
    public:
        int LayDiemToan();
        void GanDiemToan
            (int diem);

};
```

```
// file HocSinh.cpp
int CHocSinh::LayDiemToan()
{
    return mDiemToan;
}
void CHocSinh::GanDiemToan
    (int diem)
{
    if(diem >=0 && diem<=10)
        mDiemToan = diem;
}
```



# Bài tập

## Bài 2.1

---

Xác định thuộc tính và phương thức thông thường cho các lớp biểu diễn các đối tượng sau.  
Vẽ sơ đồ UML minh họa

- Phân số
- Hỗn số
- Điểm (trong không gian Oxy, Oxyz)
- Đường tròn
- Tam giác



## Bài 2.2

---

Viết lớp biểu diễn phân số và hoàn thành các phương thức sau:

- Nhập xuất phân số
- Cộng, trừ, nhân, chia hai phân số
- Rút gọn, nghịch đảo phân số
- Lấy giá trị của tử số, mẫu số
- Gán giá trị cho tử số, mẫu số
- So sánh hai phân số



# Lớp CPhanSo

---

## ● Ví dụ:

*// file PhanSo.h*

```
class PhanSo
{
private:
    int mTuSo;
    int mMauSo;
public:
    PhanSo cong(PhanSo p);
};
```

*// file PhanSo.cpp*

```
PhanSo PhanSo::cong(PhanSo p)
{
    // Cài đặt cộng phân số...
}
```

## Bài 2.3

---

- Dựa vào lớp phân số từ bài 2.2, viết chương trình cho phép
  - nhập vào một mảng các phân số, sau đó sắp xếp mảng tăng dần.
  - Xuất ra các phân số trong mảng có tử và mẫu đều là số chẵn





## Bài 2.4

---

Thông tin một học sinh bao gồm:

- Họ tên.
- Điểm văn, toán.

Viết chương trình cho phép thực hiện các thao tác trên kiểu **học sinh**:

- Nhập, xuất thông tin học sinh.
- Tính điểm trung bình.
- Xếp loại theo tiêu chí
  - Giỏi ( $\geq 8.0$ ), Khá ( $\geq 7.0$ ).
  - Trung bình ( $\geq 5.0$ ), Yếu ( $< 5$ ).



# Bài tập về nhà

---

- Làm bài 2.2, 2.3
- Deadline: 23h55 thứ 5 ngày 26/9/2013



# Tham khảo

---

- Slide bài giảng PPLTHDT của thầy
  - Đinh Bá Tiến
  - Nguyễn Minh Huy

