

Khái niệm cơ bản về Đối tượng và Lớp đối tượng



Nội dung

- Đối tượng và lớp.
- Tầm vực.
- Nạp chồng hàm.
- Bài tập.





Nội dung

- **Đối tượng và lớp.**
- **Tầm vực.**
- **Nạp chồng hàm.**
- **Bài tập.**





Khái niệm đối tượng và lớp

- Đối tượng là gì?
 - Chương trình là “cỗ máy” phức tạp.
 - Cấu thành từ nhiều loại “vật liệu”.
 - Vật liệu cơ bản: hàm, cấu trúc.
 - **Đã đủ tạo ra chương trình tốt?**



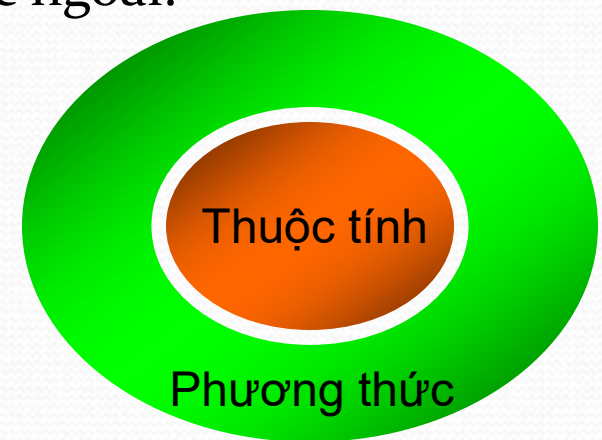
Vật liệu mới: Đối tượng!!





Khái niệm đối tượng và lớp

- Đặc trưng của đối tượng:
 - Đóng gói cả dữ liệu và xử lý.
 - Thuộc tính (attribute): dữ liệu của đối tượng.
 - Phương thức (method): xử lý của đối tượng.
- Cấu trúc đối tượng:
 - Hộp đen: thuộc tính trong, phương thức ngoài.
 - Bốn nhóm phương thức:
 - Nhóm tạo hủy.
 - Nhóm truy xuất thông tin.
 - Nhóm xử lý nghiệp vụ.
 - Nhóm toán tử.





Khái niệm đối tượng và lớp

- Vì sao là hướng đối tượng?
 - Hướng thủ tục (procedural oriented).
 - Lấy hành động làm trung tâm.
 - Hàm là xương sống.
 - Lặt (Rau) - Ướp (Cá)
 - Luộc (Rau) - Kho (Cá)
 - Nấu (Cơm)
 - Hướng đối tượng (object oriented).
 - Lấy dữ liệu làm trung tâm.
 - Đối tượng là xương sống.
 - Rau.Lặt - Cá.Ướp
 - Rau.Luộc - Cá.Kho
 - Cơm.Nấu

Các bước nấu ăn	
Verb	Object
Lặt	Rau
Ướp	Cá
Nấu	Cơm
Kho	Cá
Luộc	Rau

**Thay đổi
tư duy
lập trình!!**





Khái niệm đối tượng và lớp

- Khái niệm lớp:

Person1:

- Name: Peter.
- Age: 25.
- Hair Color: Brown.
- Eye Color: Brown.
- Job: Worker.



Person2:

- Name: Thomas.
- Age: 50.
- Hair Color: White.
- Eye Color: Blue.
- Job: Teacher.



Tập hợp đối tượng có cùng thuộc tính và phương thức

Human:

- Name.
- Age.
- Hair Color.
- Eye Color.
- Job.



**Bản mô tả đối tượng
Kiểu của đối tượng**





Khái niệm đối tượng và lớp

- Sử dụng đối tượng trong C++:
 - Khai báo lớp: file .h

```
class <Tên lớp>
{
    <Khai báo thuộc tính>;
    <Khai báo phương thức>;
};
```
 - Cài đặt phương thức: file .cpp
 - Như cài đặt hàm, kèm theo tên lớp.
 - Tạo đối tượng:
 - Như khai báo biến, kiểu dữ liệu là lớp.
 - Truy xuất thành phần đối tượng:
 - Như truy xuất thành phần cấu trúc.





Khái niệm đối tượng và lớp

- Ví dụ:

// file PhanSo.h

```
class PhanSo
{
private:
    int m_iTuSo;
    int m_iMauSo;
public:
    PhanSo cong(PhanSo p);
};
```

// file PhanSo.cpp

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

// file main.cpp

```
void main()
{
    PhanSo p1;
    PhanSo *p2 = new PhanSo;
    PhanSo *p3 = new PhanSo[10];

    p3[1] = p1.cong(p3[5]);
    p3[1] = p2->cong(p3[5]);
}
```





Nội dung

- Đối tượng và lớp.
- **Tầm vực.**
- Nạp chồng hàm.
- Bài tập.





Tầm vực

- Khái niệm tầm vực:
 - Tầm ảnh hưởng, phạm vi hoạt động của thuộc tính và phương thức.
 - Tầm ảnh hưởng càng hẹp, truy xuất càng hạn chế.

- Phân loại tầm vực:

Tầm vực	Tầm ảnh hưởng	Phạm vi hoạt động
private	Hẹp	Bên trong lớp.
public	Rộng	Bên trong lẫn bên ngoài lớp.
protected	Vừa	Bên trong lớp và lớp kế thừa.





Tâm vực

- Dr. Guru khuyên:
 - Quy tắc hộp đen:
 - Thuộc tính có tầm vực private để hạn chế truy xuất.
 - Phương thức có tầm vực public để cung cấp tính năng.

```
class PhanSo
{
private:
    int m_iTuSo;
    int m_iMauSo;
public:
    void khoiTao(int iTuSo, int iMauSo);
    PhanSo cong(PhanSo p);
};
```



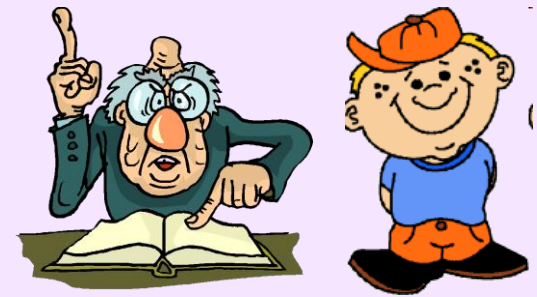


Tâm vực

- Toán tử :: (bốn chấm):
 - Lớp 5A và 5B đều có sinh viên Lê Phương.
 - **Làm sao để phân biệt?**
 - Trường hợp 1: đứng trong lớp 10A1.
 - Trường hợp 2: đứng ngoài 2 lớp.
 - **Toán tử ::**
 - Dùng gọi tên thành phần của lớp từ bên ngoài.
 - <Tên lớp>::<Tên thành phần>



Lớp 5A



Lớp 5B





Tầm vực

- Ví dụ:

```
class PhanSo
{
    private:
        int m_iTuSo;
        int m_iMauSo;
    public:
        PhanSo cong(PhanSo p)
        {
            // Cài đặt bên trong lớp.
        }
};
```

```
PhanSo PhanSo::cong(PhanSo p)
{
    // Cài đặt bên ngoài lớp.
}
```





Tâm vực

- Con trỏ this:
 - Sử dụng bên trong lớp.
 - Đại diện cho đối tượng đang gọi phương thức.
 - Hữu dụng trong một số trường hợp.

```
class PhanSo
{
private:
    int    m_iTuSo;
    int    m_iMauSo;
public:
    void ganTuSo(int iTuSo) { this->m_iTuSo = iTuSo; }
};
```

```
void main()
{
    PhanSo p1;
    p1.ganTuSo(3);

    PhanSo p2;
    p2.ganTuSo(5);
}
```





Nội dung

- Đối tượng và lớp.
- Tầm vực.
- **Nạp chồng hàm.**
- Bài tập.





Nạp chồng hàm

- Chữ ký hàm (Function Signature):

- Phân biệt giữa các hàm.
- Đặt trưng của một hàm:
 - Tên hàm.
 - Danh sách tham số.

```
double sapXep(int aMang[], int iKichThuoc);  
PhanSo PhanSo::cong(PhanSo p);
```

- **Giá trị trả về**

**Chương trình có thể có
nhiều hàm cùng tên!!**





Nạp chồng hàm

- Những trường hợp nạp chồng hàm nào không hợp lệ?
 1. `int tinhToan(int a, int b);`
 2. `int tinhToan(int x, int y);`
 3. `int tinhToan(int a, float b);`
 4. `float tinhToan(int u, int v);`
 5. `int tinhToan(int a, long b);`





Tóm tắt

- Khái niệm đối tượng và lớp:
 - Đối tượng: một thành phần tạo nên chương trình.
 - Đối tượng bao gồm thuộc tính và phương thức.
 - Lớp: tập hợp đối tượng cùng thuộc tính phương thức.
- Tầm vực:
 - Phạm vi hoạt động của thuộc tính và phương thức.
 - Có 3 mức: public, private, protected.
- Nạp chồng hàm:
 - Nhiều hàm cùng tên, khác tham số.





Nội dung

- Đối tượng và lớp.
- Tầm vực.
- Nạp chồng hàm.
- **Bài tập.**



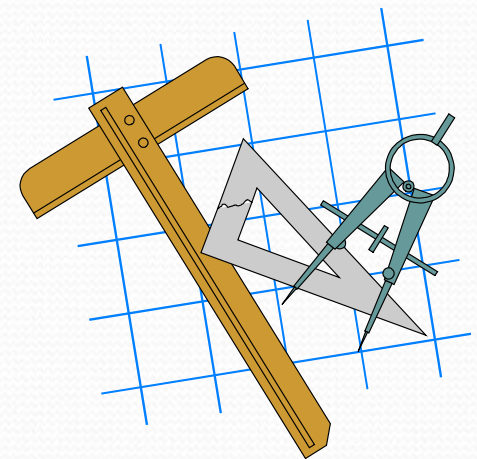


Bài tập

- Bài tập 2.1:

Xây dựng lớp **phân số** cho phép thực hiện các thao tác:

- Nhập, xuất.
- Lấy tử số, mẫu số.
- Gán giá trị cho tử số, mẫu số.
- Nghịch đảo, rút gọn.
- Cộng, trừ, nhân, chia, so sánh với phân số khác.



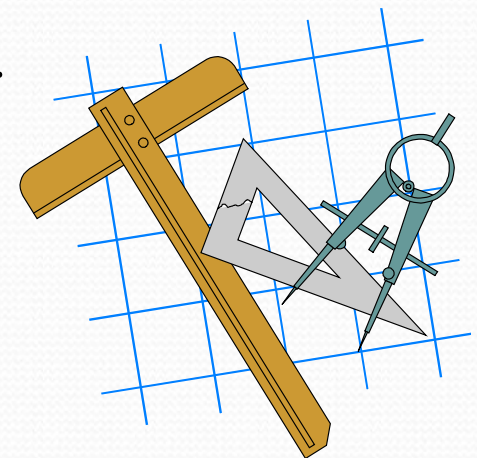


Bài tập

- Bài tập 2.2:

Xây dựng lớp **số phức** cho phép thực hiện các thao tác:

- Nhập, xuất.
- Lấy phần thực, phần ảo.
- Gán giá trị cho phần thực, phần ảo.
- Tính module.
- Cộng, trừ, nhân, chia, so sánh với số phức khác.



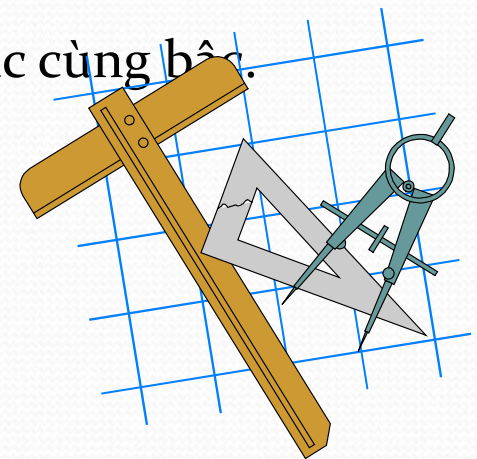


Bài tập

- Bài tập 2.3:

Xây dựng lớp **đơn thức** cho phép thực hiện các thao tác:

- Nhập, xuất.
- Lấy hệ số, số mũ.
- Gán giá trị cho hệ số, số mũ.
- Tính giá trị, đạo hàm, nguyên hàm.
- Cộng, trừ, nhân, chia, so sánh với đơn thức khác cùng bậc.





Bài tập

- Bài tập 2.4:

Thông tin một học sinh bao gồm:

- Họ tên.
- Điểm văn, toán.

Xây dựng lớp **học sinh** cho phép thực hiện các thao tác:

- Nhập, xuất.
- Lấy họ tên, điểm văn, toán.
- Gán giá trị cho họ tên, điểm văn, điểm toán.
- Tính điểm trung bình.
- Xếp loại theo tiêu chí:
 - Giỏi (≥ 8.0), Khá (≥ 7.0).
 - Trung bình (≥ 5.0), Yếu (< 5).

