

Trường Đại học Công nghiệp Thực phẩm Tp. Hồ Chí Minh

Bộ môn Công nghệ Phần mềm



Lập trình hướng đối tượng

8/28/2014

ĐẠI HỌC CHÍNH QUY

Vũ Văn Vinh



NỘI DUNG HỌC PHẦN

- ✓ Những cơ sở của ngôn ngữ lập trình C#
- ✓ Lớp và đối tượng
- ✓ Nạp chồng toán tử
- ✓ Kế thừa và đa hình
- ✓ Interface
- ✓ Array, List



ĐÁNH GIÁ HỌC PHẦN

- Tiểu luận: 20%
 - 02 bài kiểm tra trên lớp
 - Chuyên cần
- Thi giữa kỳ: 30%
 - Thi tự luận: 60 phút
 - Nội dung: chương 1, 2, 3
- Thi cuối kỳ: 50%
 - Thi tự luận: 60 phút
 - Nội dung: chương 4, 5, 6



CÁC PHƯƠNG PHÁP LẬP TRÌNH

- ✓ Lập trình tuần tự (lập trình không cấu trúc)
- ✓ Lập trình có cấu trúc
- ✓ Lập trình hướng đối tượng

■ Là phương pháp xuất hiện đầu tiên

- ❑ Các ngôn ngữ như Assembly, Basic
- ❑ Sử dụng các biến tổng thể
- ❑ Lạm dụng lệnh GOTO

■ Các nhược điểm

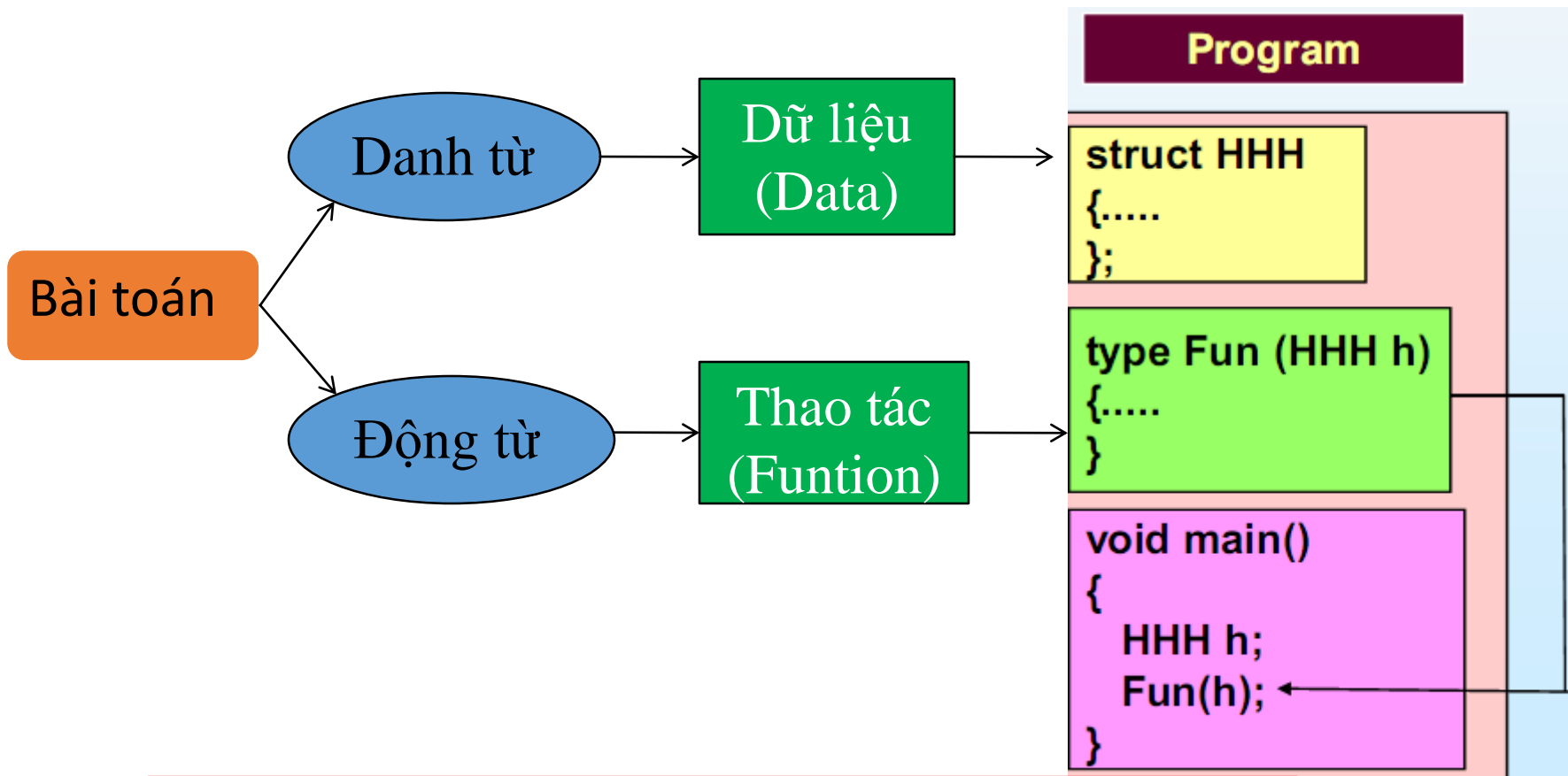
- ❑ Khó hiểu, khó bảo trì, hầu như không thể sử dụng lại
- ❑ Chất lượng kém
- ❑ Chi phí cao
- ❑ Không thể phát triển các ứng dụng lớn



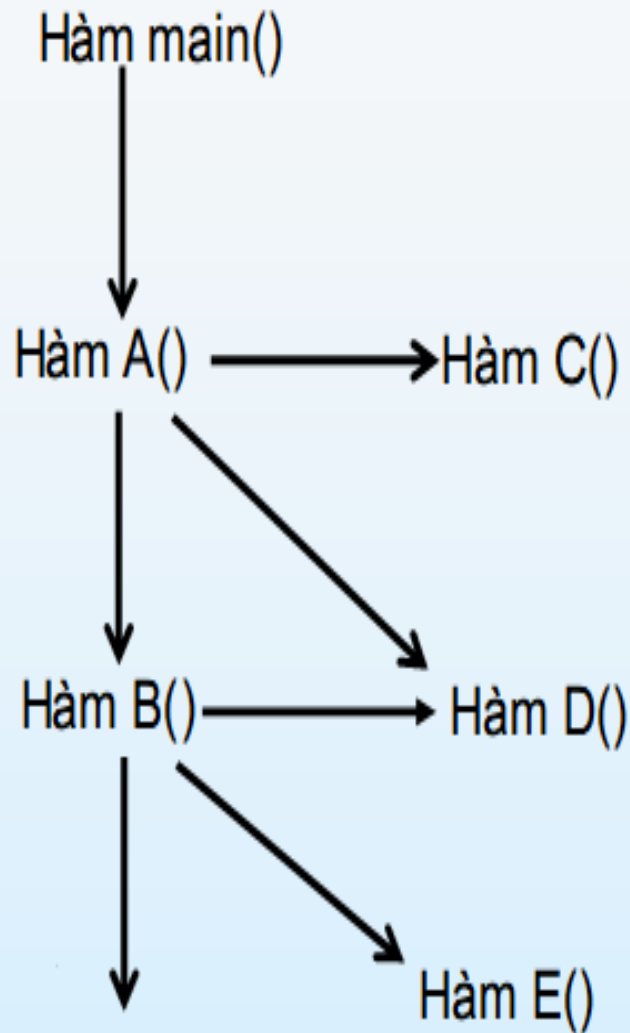
LẬP TRÌNH CÓ CẤU TRÚC

- Tổ chức thành các chương trình con (module)
- Mỗi chương trình con xử lý một phần công việc của hệ thống.
- Mỗi chương trình con lại có thể chia nhỏ thành các chương trình con nhỏ hơn
- Sử dụng các lệnh có cấu trúc: for, do while, if then else...
- Các ngôn ngữ: Pascal, C, ...
- Chương trình là tập các hàm/thủ tục

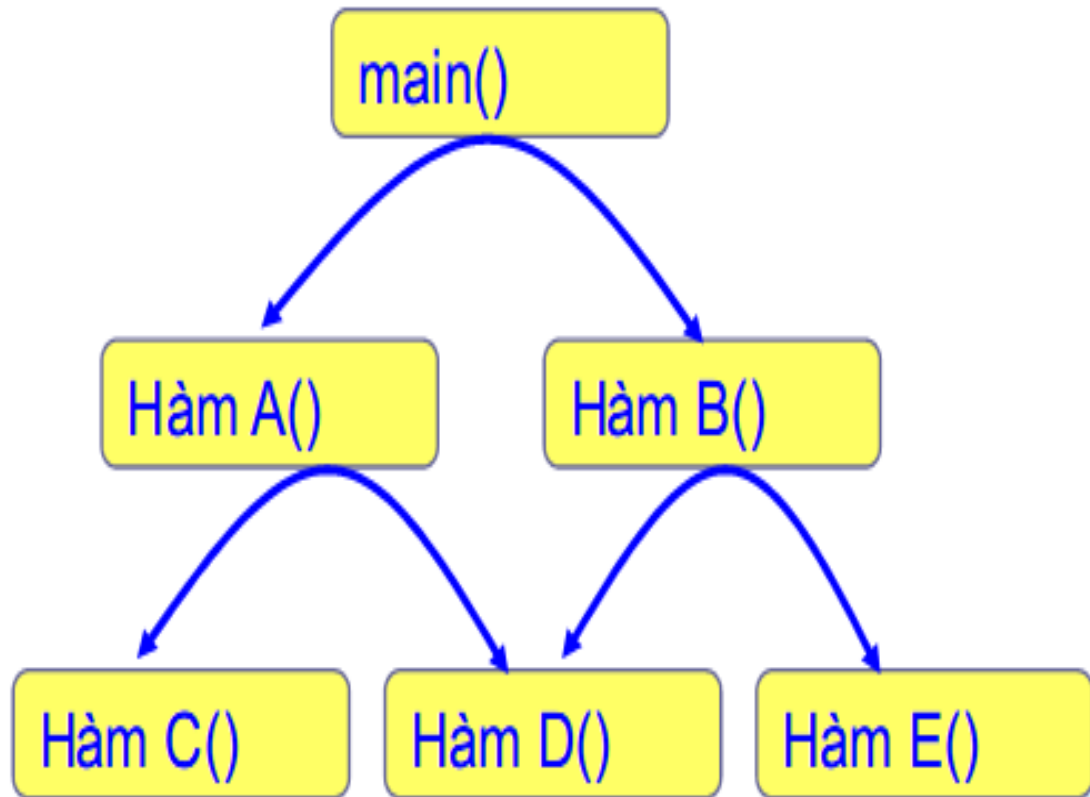
LẬP TRÌNH CÓ CẤU TRÚC



Cấu trúc DL + Giải thuật = Chương trình



Chương trình



■ Ưu điểm

- ❑ Chương trình được cục bộ hóa, do đó dễ hiểu, dễ bảo trì hơn
- ❑ Dễ dàng tạo ra các thư viện phần mềm

■ Nhược điểm

- ❑ Khi thay đổi cấu trúc dữ liệu thì mã xử lý phải thay đổi theo
- ❑ Không tự động tạo hay giải phóng dữ liệu
- ❑ Khó mô tả những quan hệ phức tạp của tự nhiên.
Biểu diễn cấu trúc “phức tạp”



LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG(OOP)

- *Lập trình hướng đối tượng (Object Oriented Programming – OOP) là một cách nhìn mới, định hướng tư duy mới, phong cách mới trong lập trình. Chương trình là sự hoạt động của các đối tượng tương tự như hoạt động của thế giới thực.*
- Lấy **đối tượng** làm nền tảng để xây dựng giải thuật, xây dựng chương trình



LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG(OOP)

- Đặt trọng tâm vào các đối tượng
- Chương trình được xem như là tập hợp các đối tượng và các đối tượng trao đổi với nhau thông qua việc nhận và gửi thông điệp (message)
- Tính mở cao
- Cơ chế bao đóng, che dấu tạo ra sự an toàn
- Hỗ trợ mạnh nguyên lý sử dụng lại nhiều nhất có thể và tạo ra mọi khả năng để kế thừa



CÁC KHÁI NIỆM CỦA OOP

- Đối tượng (Object)
- Lớp (Class)
- Trừu tượng hóa dữ liệu (Abstract data)
- Bao đóng/Che dấu thông tin (Encapsulation)
- Truyền thông điệp (Send message)
- Kế thừa (Inheritance)
- Đa hình (Plymorphism)

❖ **Đối tượng (Object):**

- Là các **sự vật** trong thế giới thực: Bàn, ghế, xe,...
- Là các **khái niệm** như: Khoa, phòng ban,...
- Có các thuộc tính và hành vi



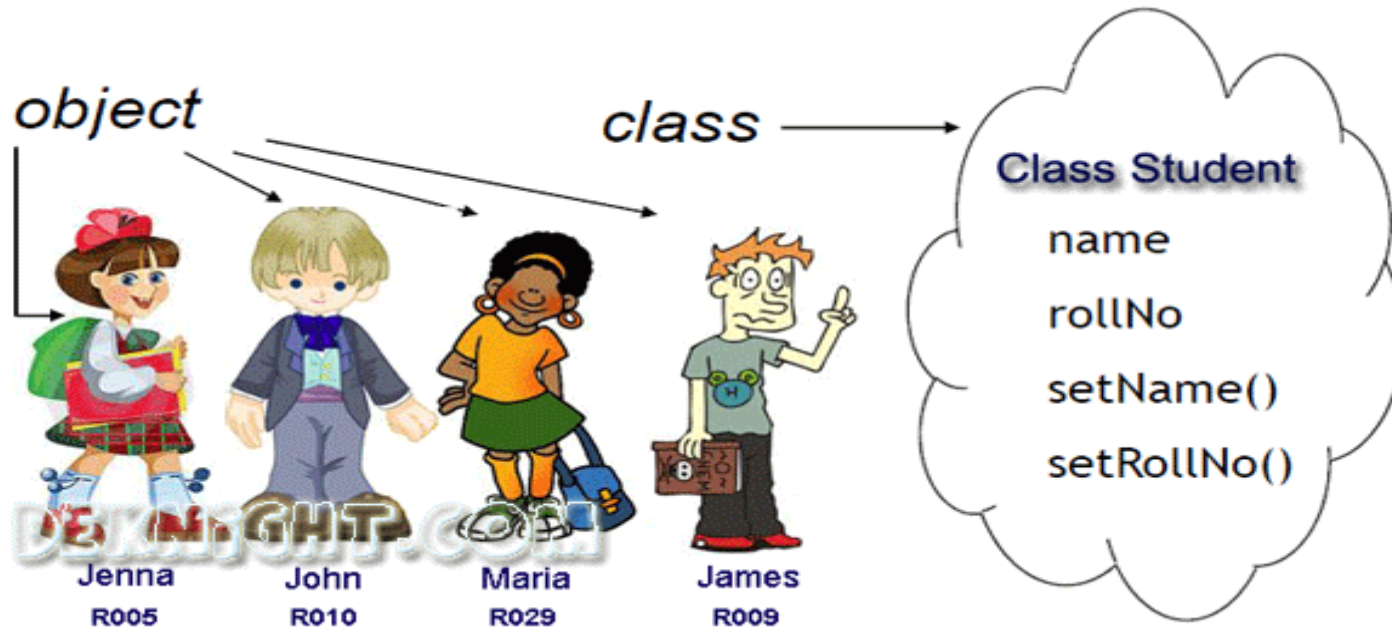


CÁC KHÁI NIỆM CỦA OOP

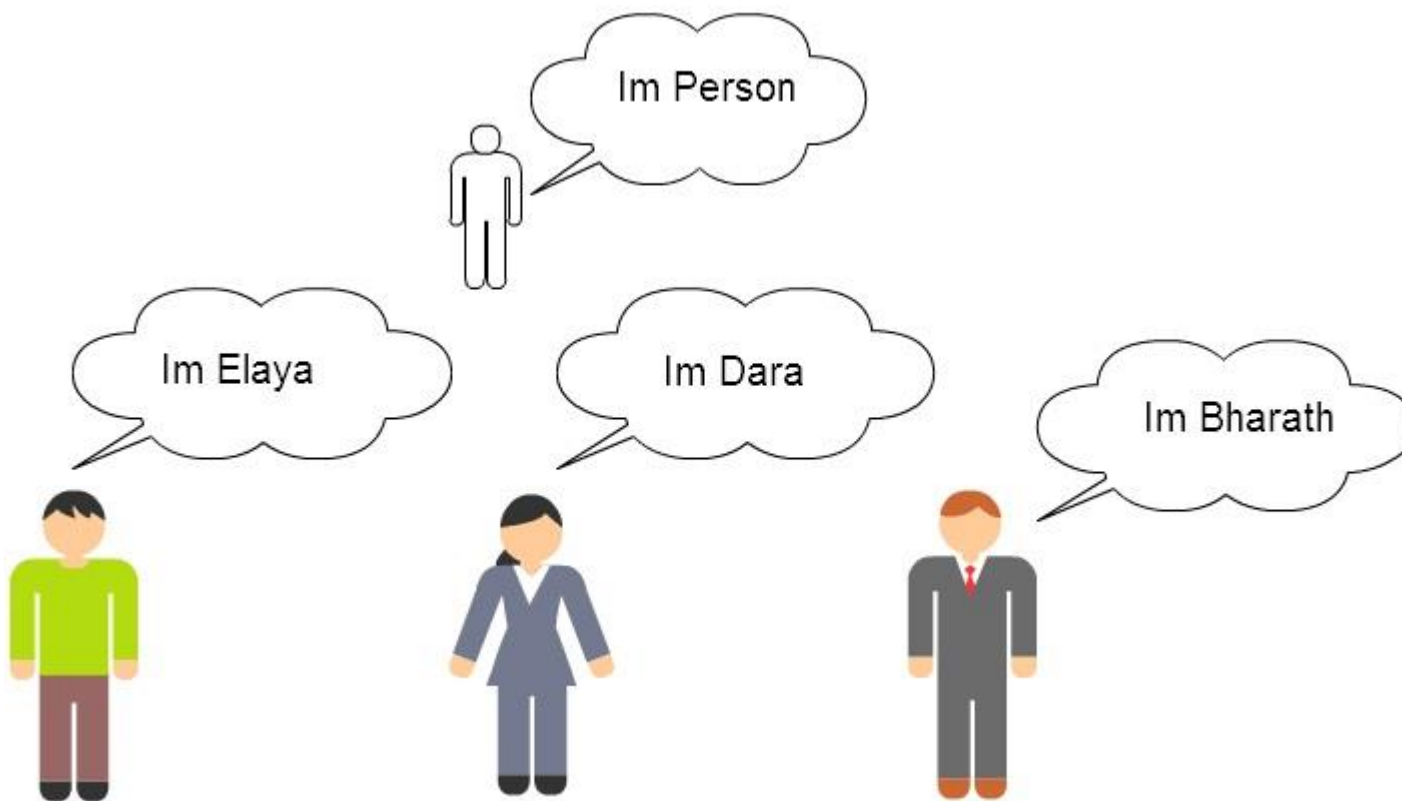
❖ Lớp (Class):

- Các đối tượng có đặc tính tương tự nhau được gom chung lại thành lớp đối tượng. Ví dụ người là một lớp đối tượng.
- Một lớp đối tượng được đặc trưng bởi các **thuộc tính** và các **hoạt động** (hành vi, thao tác)

❖ Phân biệt giữa đối tượng và lớp



❖ Phân biệt giữa đối tượng và lớp





CÁC KHÁI NIỆM CỦA OOP

❖ Các thành phần của lớp

- **Tên lớp (class name):** duy nhất, dùng phân biệt lớp này với lớp khác trong cùng một phạm vi.
- **Thành phần dữ liệu, thuộc tính (data members, attributes):** là các thành phần cấu tạo nên đối tượng, đặc trưng cho đối tượng
- **Phương thức (methods, functional members):** các hoạt động, của đối tượng thuộc lớp này

❖ Các thành phần của lớp

VD: Lớp **PhanSo** gồm các thành phần.

- **Thuộc tính:**

- Tử số
- Mẫu số

- **Phương thức :**

- Nhap(): nhập dữ liệu cho phân số
- Xuat (): xuất dữ liệu cho phân số
-



CÁC KHÁI NIỆM CỦA OOP

❖ Sơ đồ lớp

GiaoVien
MaGV
TenGV
ChuyenMon
HocVi
Giangbai
Chamthi

Sơ đồ lớp

Giáo Viên 1

010678

Nguyễn Sinh Long

Công nghệ Thông tin

Thạc sĩ

Giangbai

Chamthi

Sơ đồ đối tượng

❖ Tính trừu tượng của lớp

Là cách nhìn **đơn giản hóa** về một đối tượng mà trong đó chỉ bao gồm **những đặc điểm được quan tâm** và **bỏ qua** những chi tiết không cần thiết



Trừu tượng hóa
(Abstraction)





CÁC KHÁI NIỆM CỦA OOP

❖ Tính trừu tượng của lớp

VD: Khi lập trình biểu diễn đối tượng nhân viên:

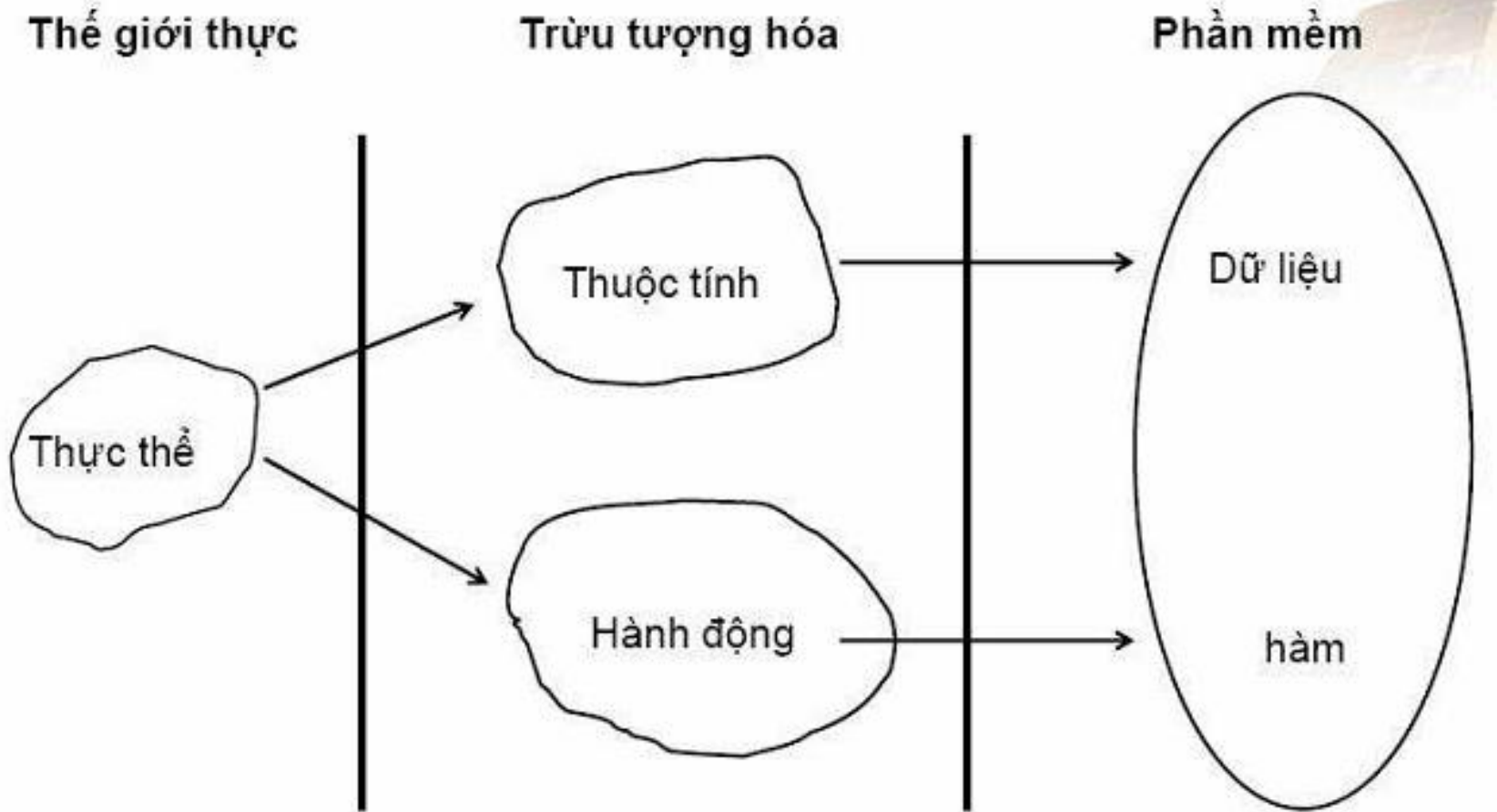
- Về thành phần dữ liệu: Không thể mô tả toàn bộ -> sử dụng các thành phần dữ liệu cần thiết
- Về thành phần hành vi: Không thể mô tả toàn bộ -> sử dụng các hoạt động liên quan tới ứng dụng

❖ Tính trừu tượng của lớp

VD: Khi lập trình biểu diễn đối tượng nhân viên:

- **Về thành phần dữ liệu:** Không thể mô tả toàn bộ -> sử dụng các thành phần dữ liệu cần thiết
- **Về thành phần hành vi:** Không thể mô tả toàn bộ -> sử dụng các hoạt động liên quan tới ứng dụng


❖ Tính trừu tượng của lớp



❖ Tính trừu tượng của lớp

Đối tượng trong thế giới thực

- Là một thực thể cụ thể mà thông thường có thể **sờ**, **nhìn thấy** hay **cảm nhận** được
- Luôn có **dữ liệu** và **hành động** (phương thức) riêng

	Dữ liệu	Hành động	
Con chó	Tên Màu Giống	Sủa Vẫy tai Chạy Ăn	

❖ Truyền thông điệp (Send Messages)

- Các đối tượng gửi và nhận thông tin với nhau giống như con người trao đổi với nhau
- Truyền thông báo cho một đối tượng là yêu cầu đối tượng đó thực hiện một công việc gì đó



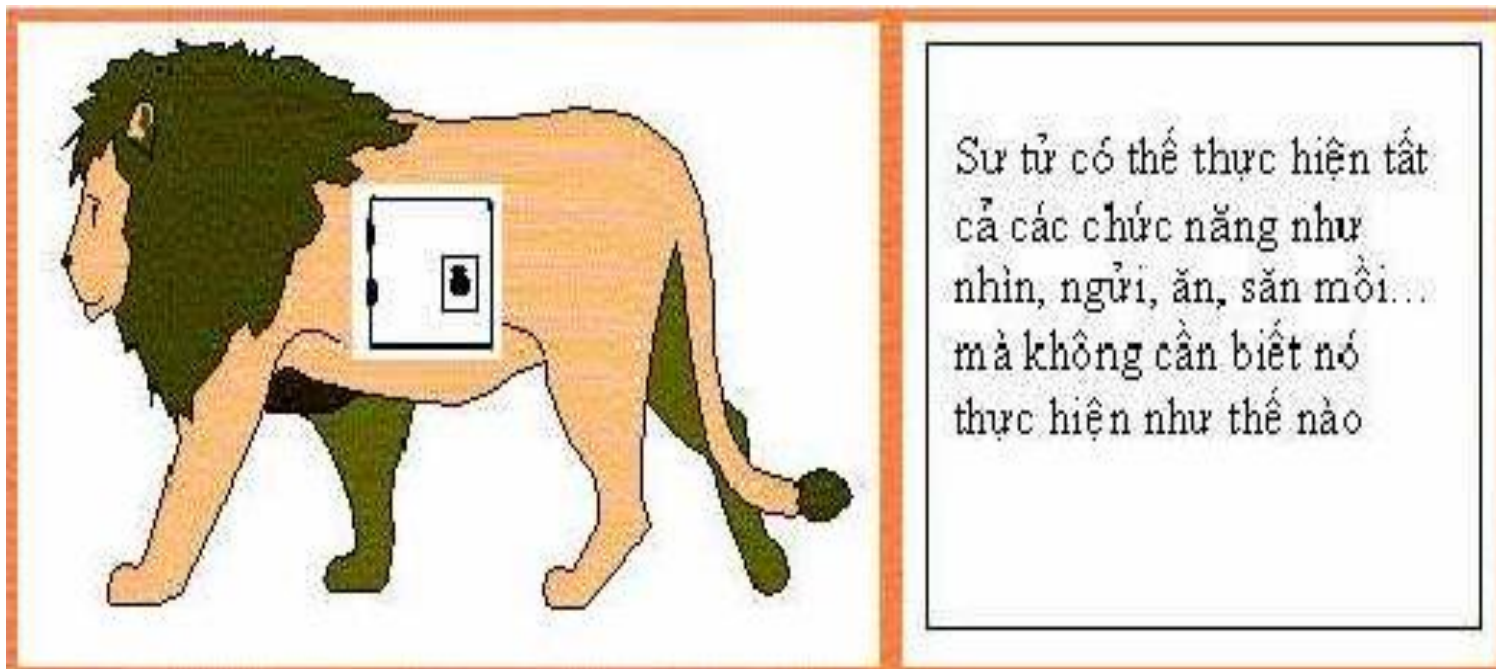


BA ĐẶC ĐIỂM CỦA OOP

- ✓ Tính bao đóng (Encapsulation)
- ✓ Tính kế thừa (Inheritance)
- ✓ Tính đa hình (Polymorphism)

❖ Tính bao đóng (Encapsulation)

- Là việc che giấu việc thực thi chi tiết của một đối tượng

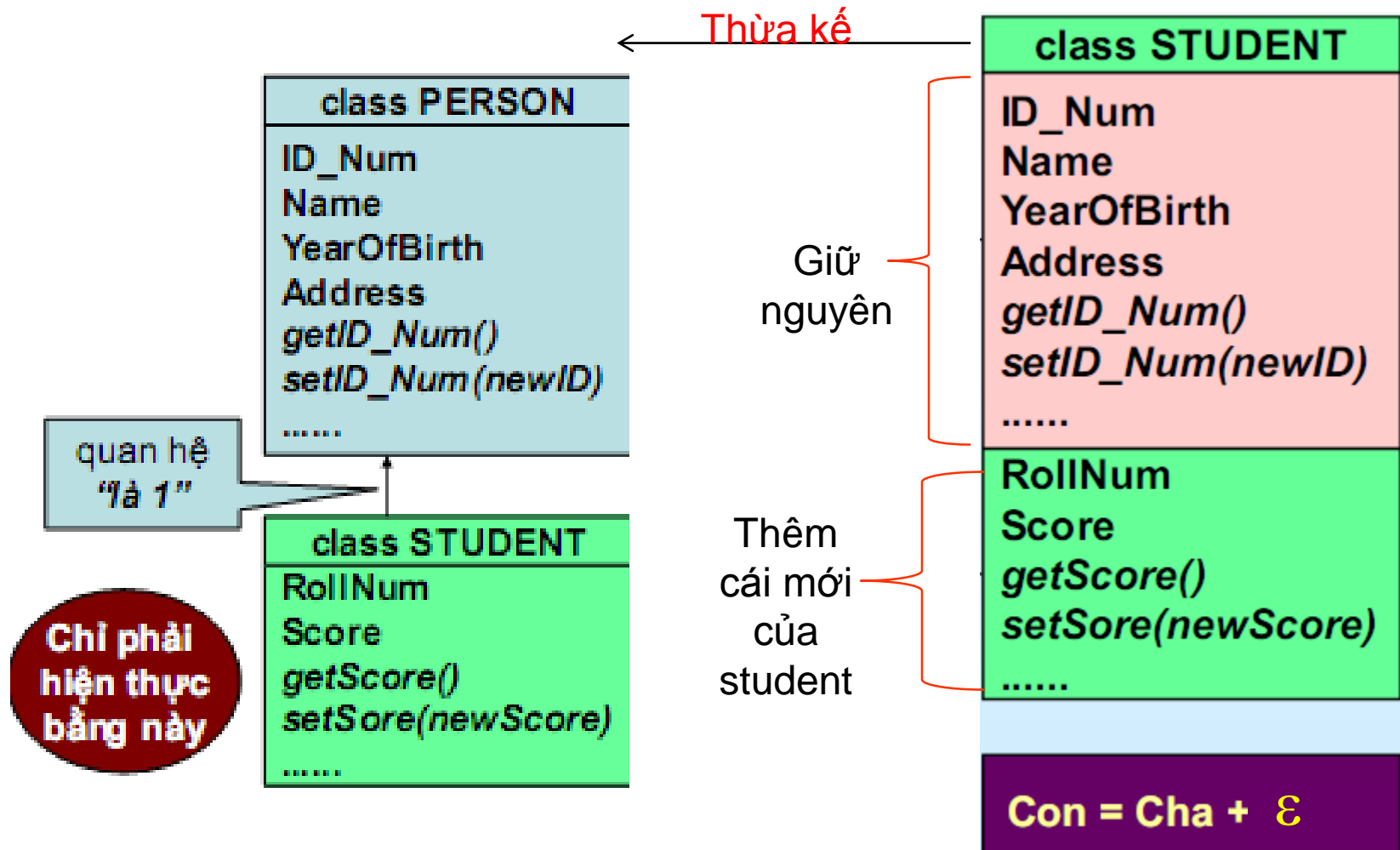


❖ Tính bao đóng (Encapsulation)

- Ẩn thông tin



❖ Tính kế thừa (Inheritance)



❖ Tính kế thừa (Inheritance)

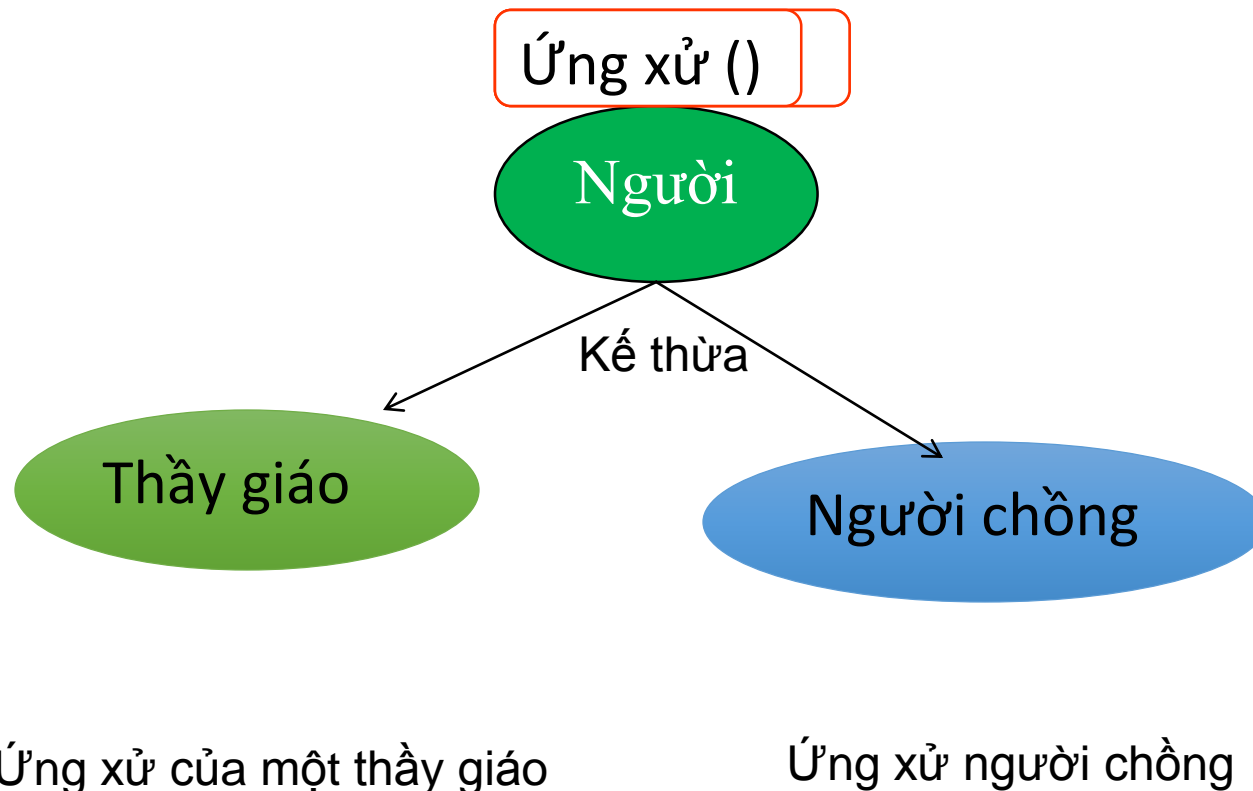
- Từ những lớp đối tượng đã có ta có thể đặc tả lớp đối tượng mới, lớp mới sẽ mang đặc tính của lớp đã có (di truyền) + đặc tính riêng của nó

Con = Cha + ε (riêng của con mà cha không có)

→ Tiết kiệm công sức viết code, test

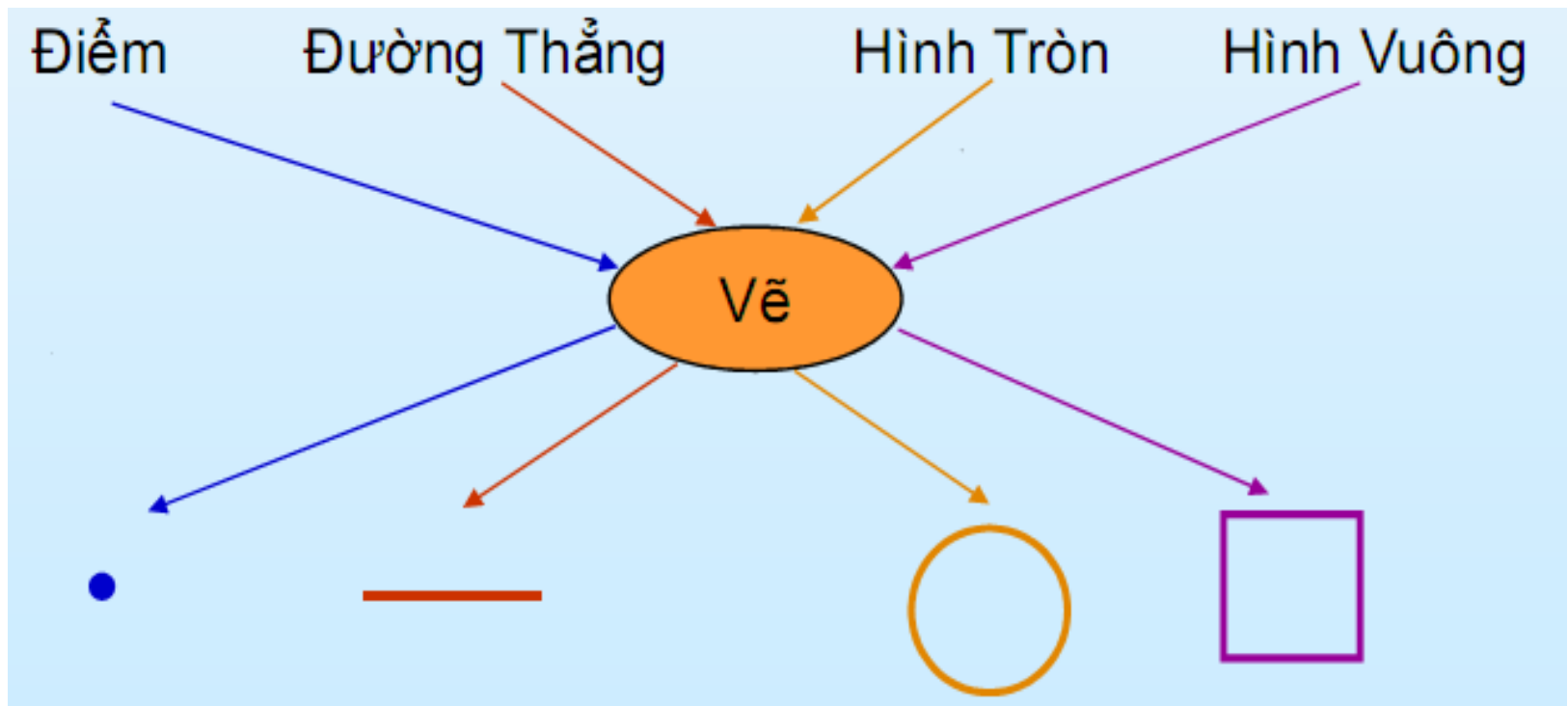
❖ Tính đa hình (Polymorphism)

- Là khả năng thực hiện khác nhau cho cùng một hành vi



❖ Tính đa hình (Polymorphism)

- Là khả năng thực hiện khác nhau cho cùng một hành vi



ƯU ĐIỂM CỦA OOP

1. Thông qua nguyên lý kế thừa, có thể loại bỏ được những đoạn chương trình lặp lại trong quá trình mô tả các lớp và mở rộng khả năng sử dụng các lớp đã được xây dựng.
2. Chương trình được xây dựng từ những đơn thể (đối tượng) trao đổi với nhau nên việc thiết kế và lập trình sẽ được thực hiện theo quy trình nhất định chứ không phải dựa vào kinh nghiệm và kỹ thuật như trước.



ƯU ĐIỂM CỦA OOP

3. Nguyên lý che giấu thông tin giúp người lập trình tạo ra được những chương trình an toàn không bị thay bởi những đoạn chương trình khác.
4. Có thể xây dựng được ánh xạ các đối tượng của bài toán vào đối tượng của chương trình.
5. Cách tiếp cận thiết kế đặt trọng tâm vào đối tượng, giúp chúng ta xây dựng được mô hình chi tiết và gần với dạng cài đặt hơn.



ƯU ĐIỂM CỦA OOP

3. Những hệ thống hướng đối tượng dễ mở rộng, nâng cấp thành những hệ lớn hơn.
4. Kỹ thuật truyền thông báo trong việc trao đổi thông tin giữa các đối tượng giúp cho việc mô tả giao diện với các hệ thống bên ngoài trở nên đơn giản hơn.
5. Có thể quản lý được độ phức tạp của những sản phẩm phần mềm.



CÁC BƯỚC THIẾT KẾ THEO OOP

1. Xác định các dạng đối tượng (lớp) của bài toán.
2. Tìm kiếm các đặc tính chung (dữ liệu chung) trong các dạng đối tượng này, những gì chúng cùng nhau chia sẻ.
3. Xác định lớp cơ sở dựa trên cơ sở các đặc tính chung của các dạng đối tượng.



CÁC BƯỚC THIẾT KẾ THEO OOP

4. Từ lớp cơ sở, xây dựng các lớp dẫn xuất chứa các thành phần, những đặc tính không chung còn lại của các dạng đối tượng.
5. Quan hệ giữa các đối tượng (Sử dụng hay kế thừa)



ỨNG DỤNG CỦA OOP

1. Những hệ thống làm việc theo thời gian thực
2. Ứng dụng mô hình hóa hoặc mô phỏng quá trình
3. Các cơ sở dữ liệu hướng đối tượng
4. Những hệ siêu văn bản, multimedia
5. Lĩnh vực trí tuệ nhân tạo và hệ chuyên gia
6. Lập trình song song và mạng nơ-ron
7. Những hệ tự động hóa và trợ giúp quyết định