

LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG **Object-Oriented Programming -** **OOP**

Bài 1: Tổng quan lập trình **Hướng đối tượng**

Trinh Thi Van Anh – PTIT

Nội dung

- Hướng dẫn môn học
- Tổng quan lập trình HĐT
- Lập trình truyền thống
- Lập trình hướng đối tượng
- Đối tượng
- Lớp
- Các khái niệm
- Phân tích và thiết kế
- Ngôn ngữ lập trình HĐT

Hướng dẫn môn học

- Thông tin môn:
 - Tên: Lập trình hướng đối tượng (Object Oriented Programming)
 - Số tín chỉ: 3
- Đánh giá môn học:
 - Điểm cc: 10% (điểm danh bất kỳ lúc nào)
 - Điểm 1: chấm trong các giờ lab (20%)
 - Điểm 2: chấm trong các giờ lab (20%)
 - Điểm thi: thi trên phòng lab (50%)

Góc trao đổi

- Group:
- **OOP-D17-E17-TrinhThiVanAnh**
- Danh sách theo dõi điểm:
- Email: vanh22@yahoo.com
Anhttv20@fe.edu.vn
- Mob: 0912 004 866

Tổng quan lập trình hướng đối tượng

- Phương pháp tiếp cận của lập trình truyền thống
- Phương pháp tiếp cận hướng đối tượng
- So sánh sự khác biệt giữa hai cách tiếp cận



Lập trình truyền thống (1)

- Lập trình tuyến tính (tuần tự)
 - Đơn giản: tuần tự
 - Đơn luồng: chỉ một luồng xử lý
 - Khó sửa lỗi, bảo trì
- Lập trình có cấu trúc (POP- Procedure Oriented Programming – Hướng thủ tục)
 - Xây dựng chương trình dựa trên các hàm/thủ tục/chương trình con
 - Chương trình sẽ gọi chương trình con theo kịch bản định trước
 - Mỗi chương trình con sẽ được triệu gọi nhiều lần
 - Chương trình con được triệu gọi bất kỳ, không theo thứ tự khai báo

Lập trình truyền thống (2)

- Chương trình = cấu trúc dữ liệu + giải thuật
(Data structure + Algorithm = Program)
- Chương trình dễ đọc dễ hiểu
- Tư duy giải thuật rõ ràng
- Khi thay đổi cấu trúc thì giải thuật cũng thay đổi theo
- Phù hợp với phạm vi trong mỗi module, không phù hợp với chương trình có nhiều module, gọi module sẽ khó quản lý

Lập trình truyền thống (3)

(Hạn chế)

- Lập trình hướng cấu trúc đã rất phổ biến trong những năm 80 và đầu những năm 90. Phương pháp lập trình thủ tục hay lập trình cấu trúc thường đi đôi với phương pháp phân tích trên xuống (top-down). Theo phương pháp này, người thiết kế hệ thống chia các chức năng (hàm) chính của hệ thống thành các chức năng nhỏ hơn, đến lượt mình, các chức năng nhỏ này lại được chia tiếp thành các chức năng nhỏ hơn nữa cho đến khi được các khối (hàm) chương trình đủ nhỏ. Việc phân tích này được thể hiện trực quan theo sơ đồ khối. Nó không phù hợp với các chương trình lớn.
- Đến nay, những ngôn ngữ lập trình hướng cấu trúc chỉ còn được sử dụng để dạy học và lập trình những chương trình nhỏ mang tính chất cá nhân. Trong thương mại, nó đã không còn được dùng đến nhiều.

Đối tượng

Đối tượng là gì?

"Tất cả mọi thứ đều là đối tượng"

Alan Kay

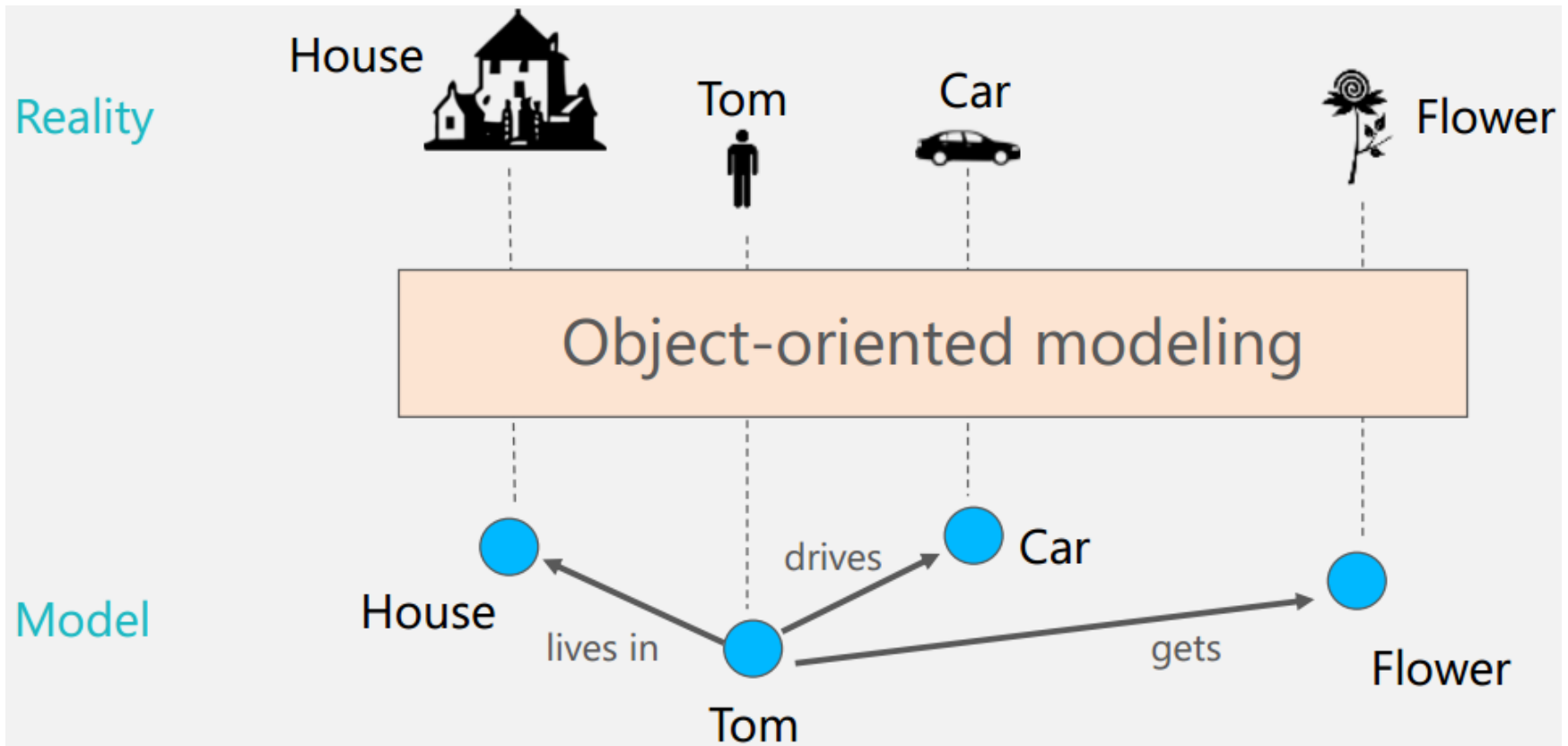


Lập trình hướng đối tượng (1)

- OOP – Object Oriented Programming.
- Chương trình là sự hoạt động của các đối tượng (Giống tự nhiên).
- Đối tượng thực thi một hoạt động tức là đối tượng thực hiện một hành vi mà đối tượng này có khả năng.
- Một chương trình là một trật tự các lời yêu cầu đối tượng thực hiện hành vi của mình.
- Đóng gói dữ liệu nên hạn chế việc truy cập tự do (private trong hướng đối tượng, chỉ các phương thức thuộc lớp mới truy cập được)
- Sử dụng lại mã nguồn, hạn chế việc viết lại mã nguồn

Lập trình HĐT (2)

- Coi chương trình phần mềm là một tập hợp các đối tượng tương tác với nhau



Lập trình HĐT (3)

- Sơ lược về OOP
 - Đối tượng (object): dữ liệu + hành vi.
 - Đối tượng phải thuộc một lớp (class).
 - Một nhóm đối tượng được biểu diễn bởi Lớp(Class)
 - Lớp= data (biến, thuộc tính) + methods (code).

Lập trình HĐT (4)

- Đặc trưng (tính chất – feature or property)
 - Trừu tượng (Abstraction)
 - Đóng gói/Che dấu thông tin (Encapsulation – Information hiding)
 - Thừa kế (Inheritance)
 - Đa hình (Polymorphism)
- Ưu điểm
 - Khi thay đổi cấu trúc dữ liệu thì không cần thay đổi mã nguồn của đối tượng khác
 - Có thể sử dụng lại mã nguồn, tiết kiệm tài nguyên
 - PP tiếp cận HĐT phù hợp với các dự án phần mềm lớn, phức tạp

PP HĐT và PP cấu trúc

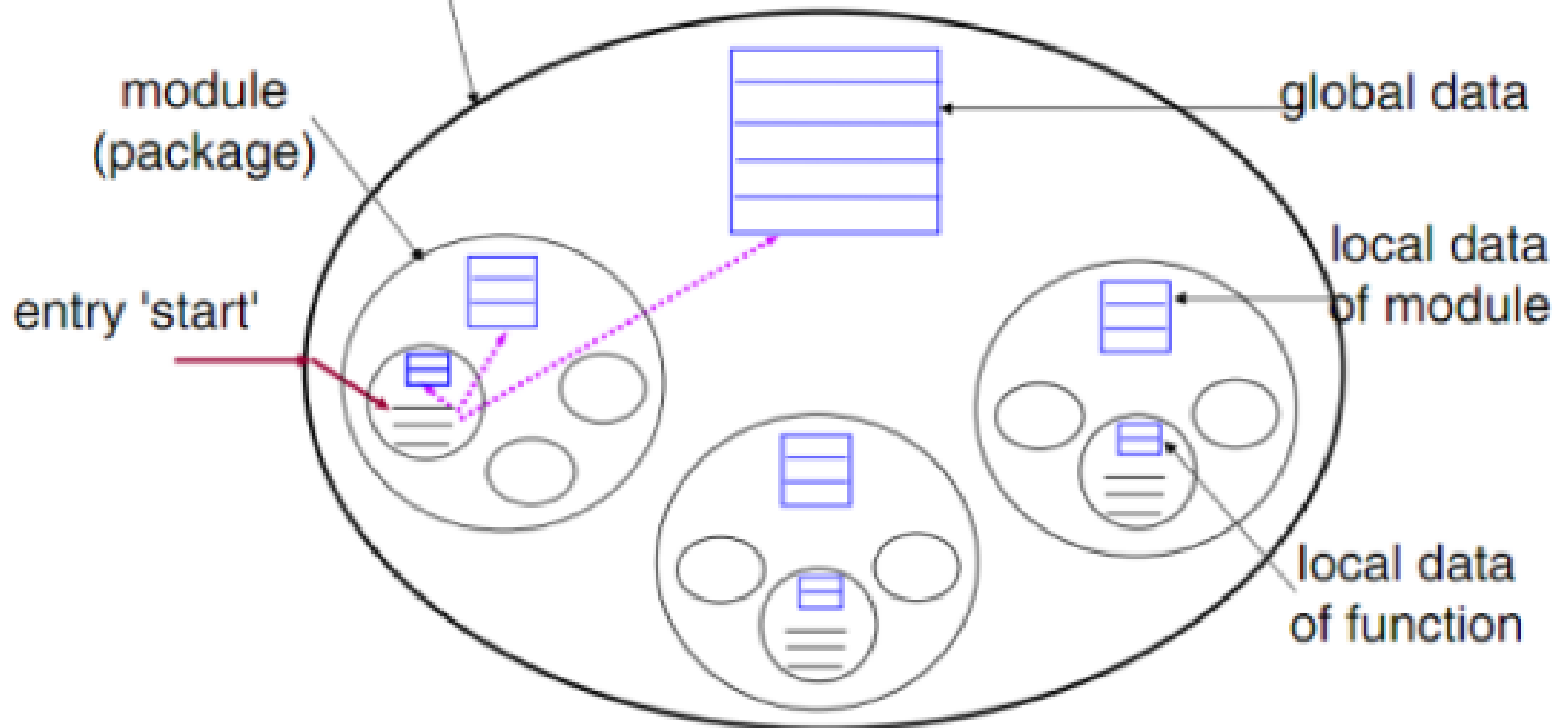
- Phương pháp
 - Phương pháp hướng đối tượng đi từ chi tiết đến trừu tượng hóa ở mức cao
 - Phương pháp cấu trúc đi từ tổng quan rồi chia nhỏ thành các bài toán con, cụ thể hơn.
- Về hạn chế truy xuất dữ liệu (đóng gói)
 - Phương pháp hướng đối tượng cho phép ẩn dữ liệu và hạn chế truy cập dữ liệu. Cho phép sử dụng lại mã nguồn để tiết kiệm tài nguyên.
 - Phương pháp cấu trúc có ràng buộc giữa cấu trúc dữ liệu và các thủ tục hoặc hàm đi kèm.

Ưu nhược điểm

- Phương pháp hướng đối tượng:
 - Hạn chế truy cập từ bên ngoài
 - Tiết kiệm tài nguyên
 - Khó theo dõi luồng dữ liệu
 - Không thiên hướng về giải thuật
- Phương pháp cấu trúc:
 - Thiên hướng về giải thuật
 - Dễ theo dõi luồng giải thuật
 - Khi thay đổi cấu trúc thường phải viết lại giải thuật
 - Chương trình đơn giản dễ hiểu

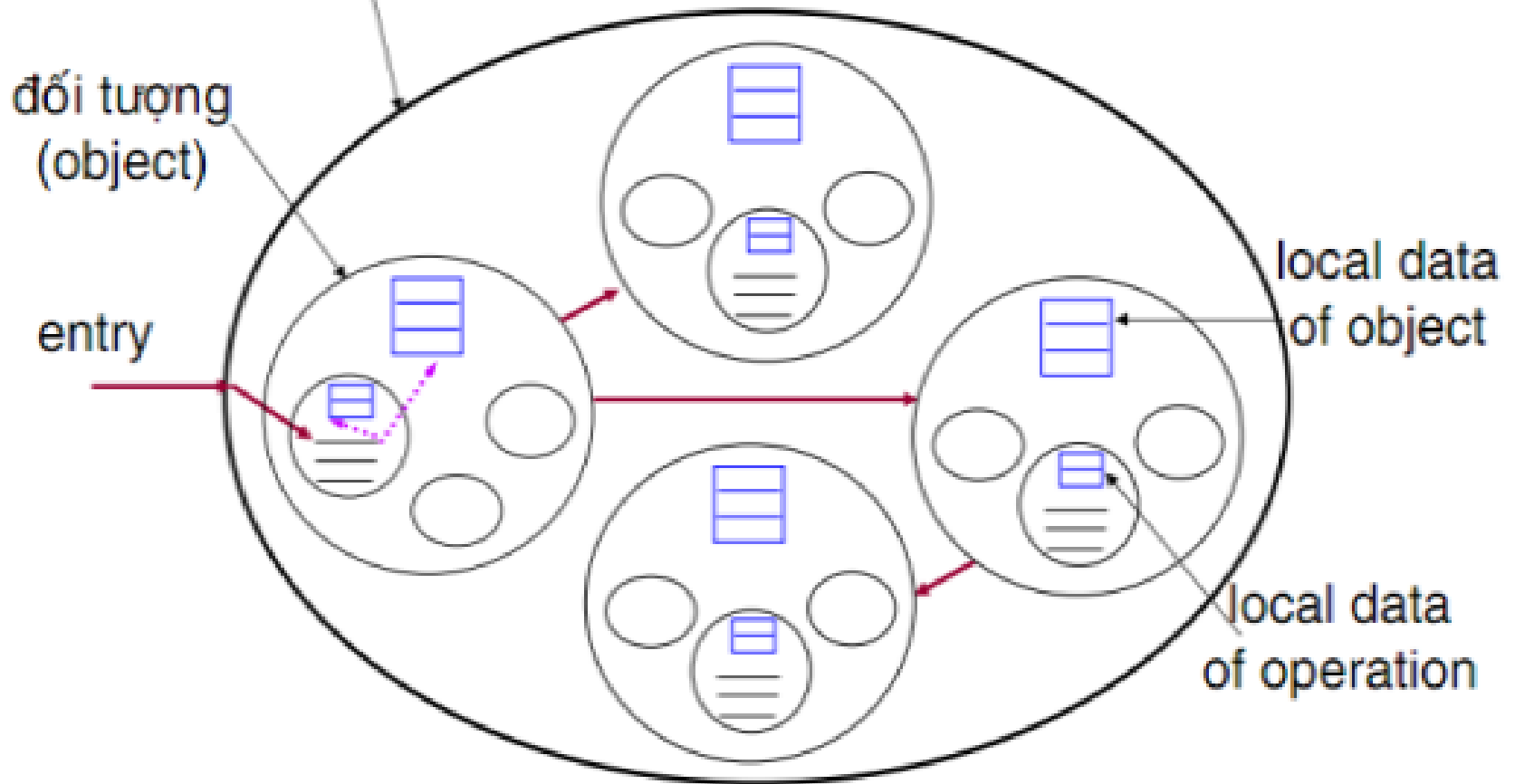
Lập trình cấu trúc

Chương trình = cấu trúc dữ liệu + giải thuật



Lập trình hướng đối tượng

Chương trình = tập các đối tượng tương tác nhau



Các khái niệm

- Đối tượng
- Lớp đối tượng
- Kế thừa
- Đóng gói
- Đa hình
- Phân tích thiết kế HĐT
- Một số ngôn ngữ lập trình HĐT

Đối tượng (1)

- Trong thế giới thực
 - Ví dụ: Xe máy, điện thoại, dịch vụ....
- Mỗi đối tượng đều có:
 - Các thông tin, trạng thái
 - Ví dụ: Các thông tin về xe máy có màu sắc, tốc độ, năm sx...
 - Các hoạt động
 - Ví dụ: Tăng ga, phanh, giảm tốc, đi đến một địa điểm...

Đối tượng (2)

	Trạng thái	Hành vi	
Con chó	Tên Màu sắc Giống chó Cảm xúc	Sủa Vẫy đuôi Ăn Chạy	
Ô tô	Hãng sản xuất Kích thước Màu sắc Giá tiền	Tăng tốc Giảm tốc Đâm	
Tài khoản	Tên tài khoản Số tài khoản Ngân hàng Số dư	Rút tiền Gửi tiền Kiểm tra số dư	

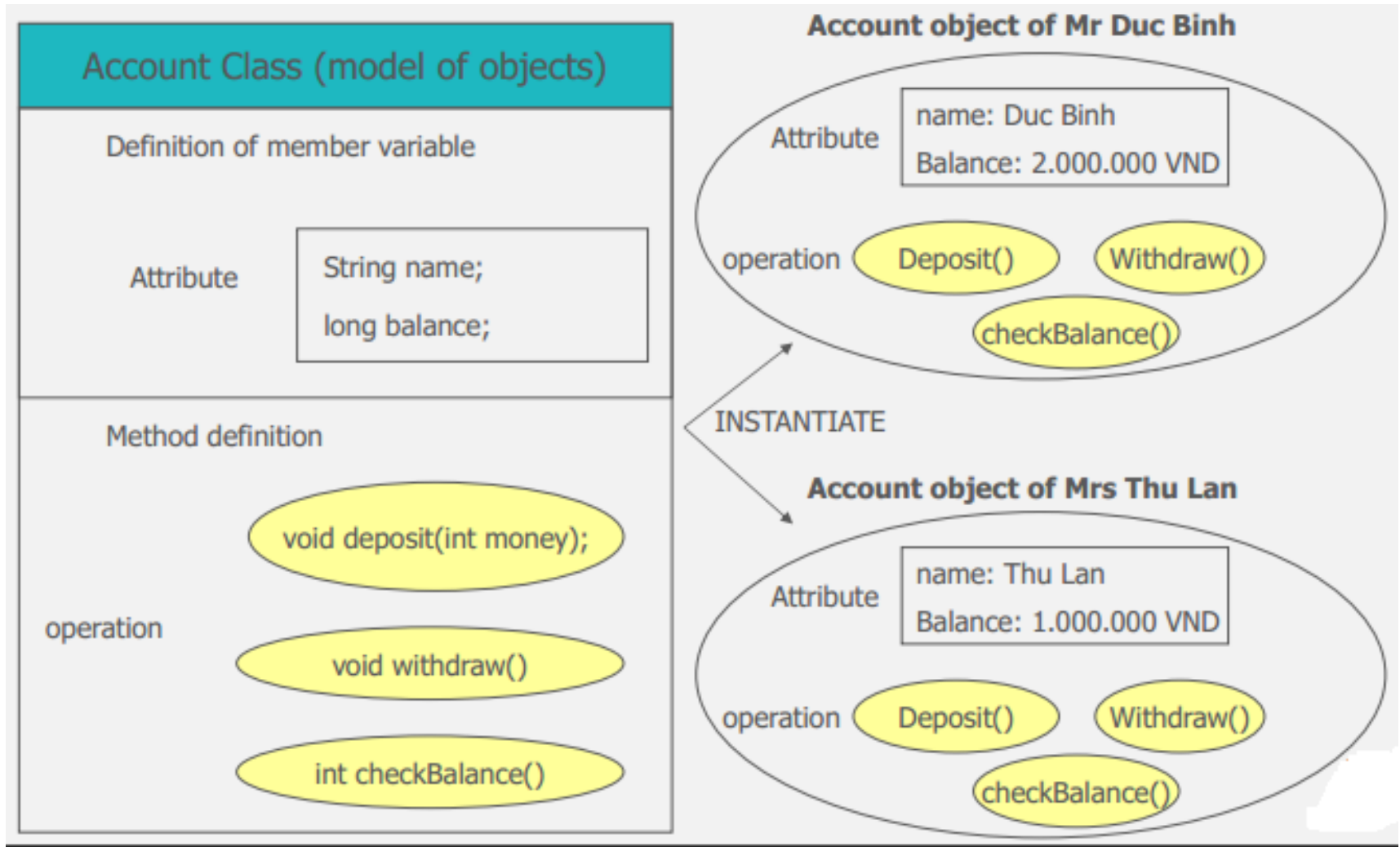
Lớp đối tượng

- Lớp là 1 khái niệm, mang tính trừu tượng, dùng để biểu diễn 1 tập các đối tượng
- Đối tượng là 1 thể hiện cụ thể của lớp, là 1 thực thể tồn tại trong hệ thống
- Lớp dùng để biểu diễn đối tượng nên lớp cũng có thuộc tính và phương thức
- Thuộc tính của lớp tương ứng với các thuộc tính của các đối tượng
- Phương thức của lớp tương ứng với các hành động của các đối tượng

Lớp và đối tượng

- Lớp là sự trừu tượng hóa của các đối tượng. Trong khi đó, đối tượng là một thể hiện của lớp
- Đối tượng là 1 thực thể cụ thể, có thực, tồn tại trong hệ thống. Trong khi đó lớp là 1 khái niệm trừu tượng, chỉ tồn tại ở dạng khái niệm để mô tả các đặc tính chung của 1 số đối tượng
- Tất cả các đối tượng thuộc về cùng 1 lớp, có cùng các thuộc tính và phương thức
- Một lớp là 1 nguyên mẫu của đối tượng. Nó xác định các hành động khả thi và các thuộc tính cần thiết cho 1 nhóm các đối tượng cụ thể.

Lớp và Đối tượng



Khái niệm kế thừa

- Cho phép dẫn xuất có thể sử dụng các thuộc tính và phương thức của lớp cơ sở tương tự như sử dụng các thuộc tính và phương thức của mình
- Cho phép việc chỉ cần cài đặt phương thức ở 1 lớp cơ sở, mà có thể sử dụng được ở tất cả các lớp lớp dẫn xuất
- Cho phép tránh dự trùng lặp mã nguồn của chương trình
- Cho phép chỉ thay đổi 1 lần khi cần thay đổi dữ liệu của 1 lớp

Đóng gói

- Cho phép che dấu sự cài đặt bên trong phương thức. Khi sử dụng chỉ việc gọi các phương thức theo 1 cách thống nhất.
- Cho phép che dấu dữ liệu bên trong đối tượng. Khi sử dụng, chúng ta không biết được thực sự bên trong đối tượng như thế nào. Chỉ dùng được những gì đối tượng cho phép.
- Cho phép hạn chế tối đa việc sửa lại mã chương trình. Khi thay đổi 1 công thức tính toán của phương thức chúng ta chỉ thay đổi mã bên trong phương thức, chứ không cần thay đổi các chương trình chứa lời gọi phương thức thay đổi.

Đa hình

- Cho phép các lớp được định nghĩa các phương thức trùng nhau: cùng tên, cùng số lượng và kiểu tham số, cùng kiểu trả về. Đây là sự nạp chồng phương thức.
- Ghi gọi các phương thức trùng tên, dựa vào đối tượng đang gọi mà chương trình sẽ thực hiện phương thức của lớp tương ứng. Do đó sẽ cho kết quả khác nhau.

Phân tích thiết kế

- Phương pháp phân tích và thiết kế
 - Phân tích: ngôn ngữ đặc tả mô hình UML
 - Thiết kế: dựa trên các mô hình phân tích, cài đặt ứng dụng/chương trình theo một ngôn ngữ lập trình hướng đối tượng.
 - Các bước phân tích thiết kế hướng đối tượng
 - Mô tả bài toán
 - Đặc tả yêu cầu
 - Trích chọn đối tượng
 - Mô hình hóa lớp đối tượng
 - Thiết kế tổng quan
 - Thiết kế chi tiết

Ngôn ngữ lập trình OOP

- C++ (Borland C++, Visual C++)
- **Java**
- C# (C sharp)
- VB.net
-
- C++, MS VC++: hỗ trợ cả POP lẫn OOP. Hỗ trợ đa thừa kế. Đối tượng là biến của chương trình. Hàm main() là POP.
- Java (Sun), C# (Microsoft): chỉ hỗ trợ OOP, hàm main phải nằm trong một lớp. Chỉ hỗ trợ đơn thừa kế.