



Môn: Lập trình Hướng đối tượng (Object Oriented Programming)

Chương 1. Tổng quan về cách tiếp cận hướng đối tượng



Môn: Lập trình Hướng đối tượng (Object Oriented Programming)

Chương 1. Tổng quan về cách tiếp cận hướng đối tượng

Số tiết LT :30t TH:30 st: 3T/tuần

Sbài KT: TK:4bài Gk : 1bài TH

Bài KTTHCK: 1bài

Bài THI CK: TH

Nội dung

- 1.1. Phương pháp tiếp cận của lập trình truyền thống
- 1.2. Phương pháp tiếp cận hướng đối tượng
- 1.3. So sánh sự khác biệt giữa hai cách tiếp cận
- 1.4. Xu hướng phát triển của lập trình hướng đối tượng

1.1. Phương pháp tiếp cận của LT truyền thống

- Lập trình tuyến tính
 - Đơn giản: tuần tự từ
 - Đơn luồng: chỉ một luồng xử lý

1.1. PP tiếp cận của LT truyền thống (tt)

Lập trình có cấu trúc

- Chia nhỏ thành 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
- **Chương trình = cấu trúc dữ liệu + giải thuật**
- 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

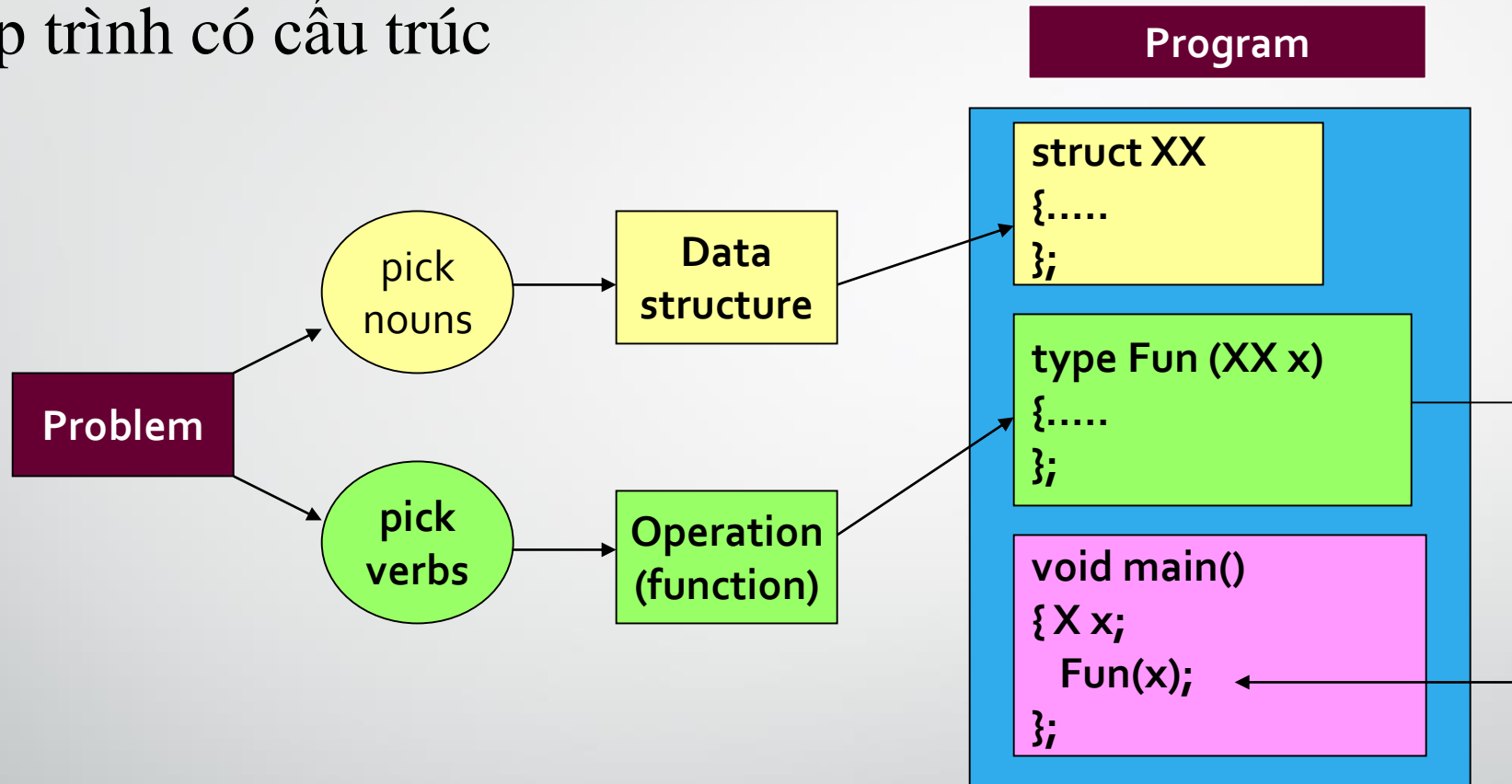
1.1. PP tiếp cận của LT truyền thống (tt)

Lập trình có cấu trúc

- Chia nhỏ thành 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
- 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

1.1. PP tiếp cận của LT truyền thống (tt)

Lập trình có cấu trúc



1.1. PP tiếp cận của LT truyền thống (tt)

Hạn chế của lập trình truyền thống

- Lập trình hướng cấu trúc đã rất phổ biến trong những năm 80 và đầu những năm 90, nhưng do những hạn chế và những nhược điểm rõ ràng khi lập trình hệ thống lớn, lập trình hướng cấu trúc đã dần bị thay thế cho phương pháp lập trình hướng đối tượng.
- Hiện 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, phương pháp lập trình truyền thống đã không còn được dùng đến nhiều.

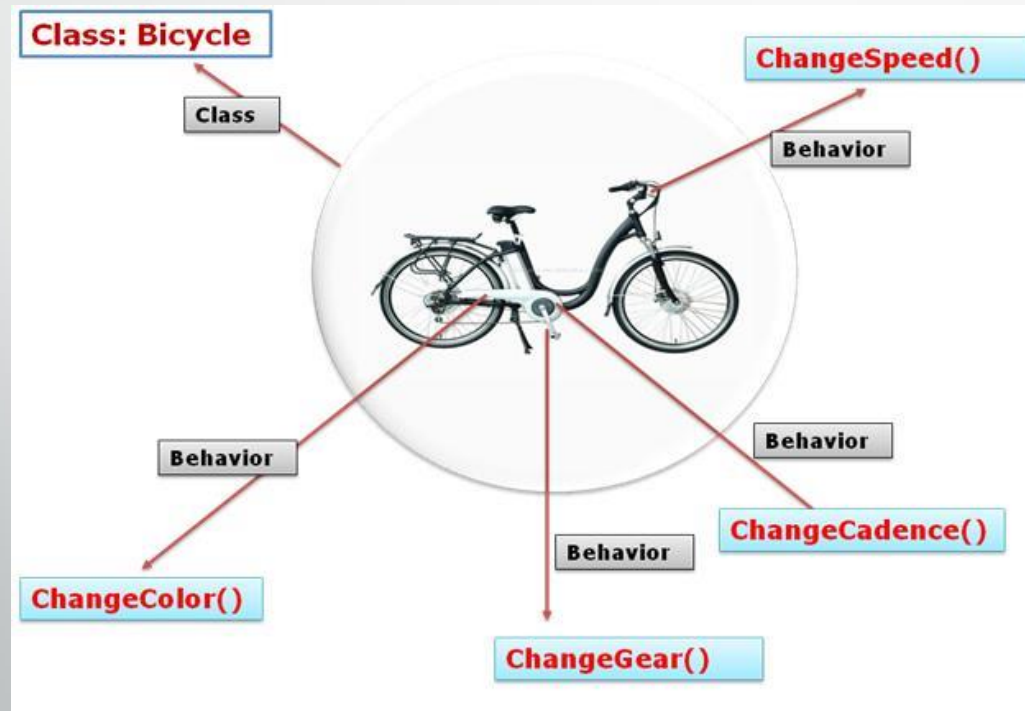
1.2. Phương pháp tiếp cận hướng đối tượng

- 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) làm không kiểm soát được việc cập nhật dữ liệu
- Sử dụng lại mã nguồn, hạn chế việc viết lại mã nguồn

1.2. Phương pháp tiếp cận hướng đối tượng (tt)

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).**



1.2. Phương pháp tiếp cận hướng đối tượng (tt)

Đặc trưng (tính chất)

- 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

1.2. Phương pháp tiếp cận hướng đối tượng (tt)

Phương pháp phân tích và thiết kế theo hướng đối tượng

- 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

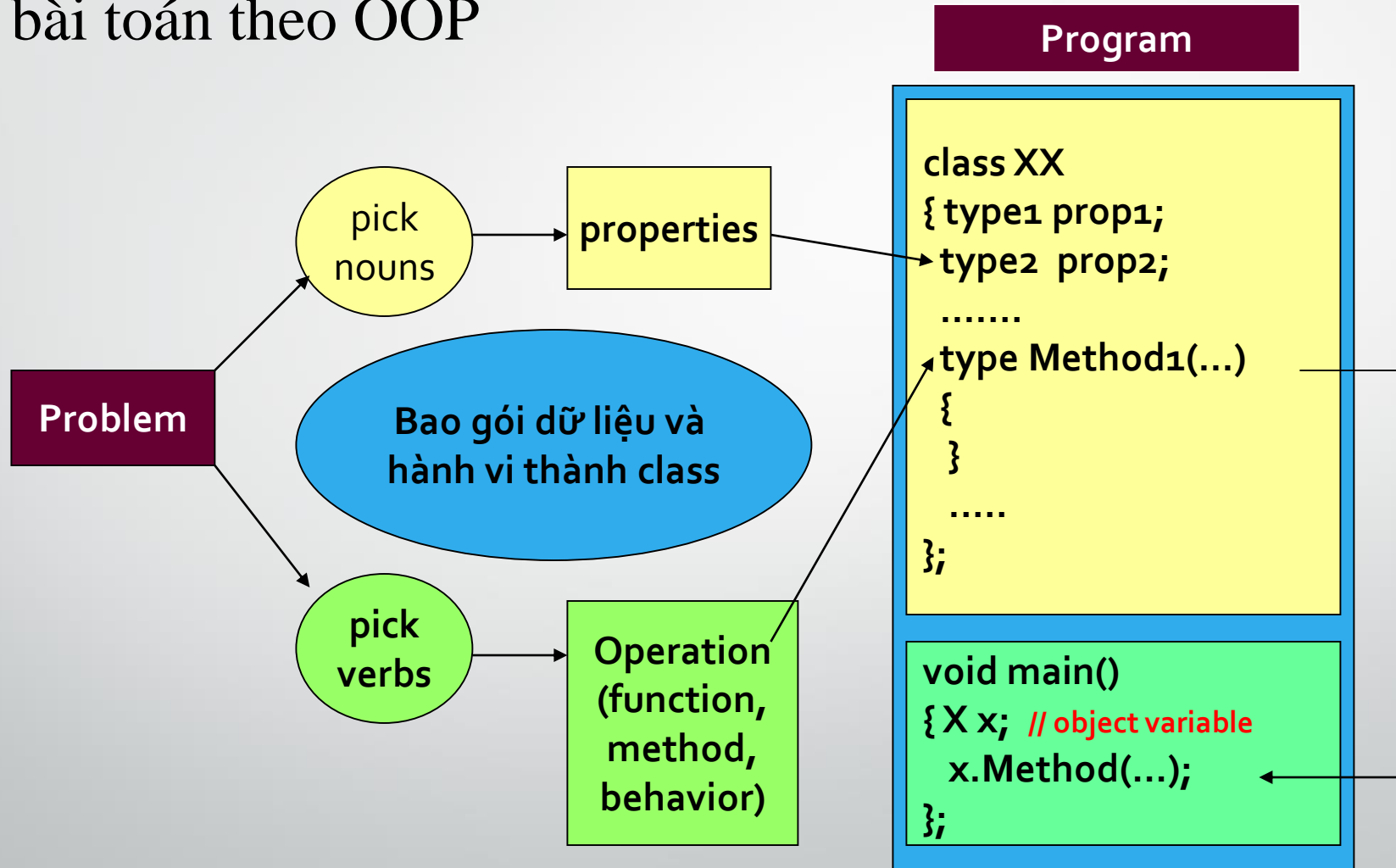
1.2. Phương pháp tiếp cận hướng đối tượng (tt)

Ngôn ngữ lập trình OOP

- C++ (Borland C++, Visual C++)
- Java
- C# (C sharp)
- Visual Basic.
-
- C++, MS VC++: hỗ trợ cả POP (lập trình cấu trúc Procedure Oriented Programming) lẫn OOP → Lai 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.

1.2. Phương pháp tiếp cận hướng đối tượng (tt)

Giải bài toán theo OOP



1.3. So sánh sự khác biệt giữa 2 cách tiếp cận

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.

1.3. So sánh sự khác biệt ... (tt)

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

1.4. Xu hướng phát triển của lập trình HĐT

- Hướng thành phần
- Hướng Agent
- Hướng Aspect

1.4. Xu hướng phát triển của LT HĐT (tt)

Hướng thành phần

- Xuất phát từ lập trình hướng đối tượng
- Giải quyết bài toán từ các thành phần có tính độc lập với nhau, mỗi thành phần đảm nhiệm một công việc nhất định
- Các thành phần được lắp ghép với nhau để thỏa các yêu cầu phần mềm
- Ưu điểm của lập trình theo hướng thành phần
 - Các lập trình viên chia sẻ các thành phần
 - Tiết kiệm công sức lập trình dựa trên các thành phần có sẵn

1.4. Xu hướng phát triển của LT HĐT (tt)

Hướng Agent

- Lập trình hướng agent giống như hướng thành phần nhưng có mức trừu tượng cao hơn.
- Các agent là các thành phần có khả năng hoạt động độc lập
- Các agent có thể chủ động liên lạc với các agent khác khi cần

1.4. Xu hướng phát triển của LT HĐT (tt)

Hướng Aspect

- Lập trình hướng aspect cũng là một xu hướng của lập trình hướng thành phần.
- Mỗi thành phần thực hiện theo luồng công việc.
- Phần mềm được xây dựng từ việc tổng hợp các luồng công việc

