

LẬP TRÌNH OOP VỚI JAVA

GIỚI THIỆU LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

ThS. Dương Hữu Thành
Khoa CNTT, Đại học Mở Tp.HCM
thanh.dh@ou.edu.vn



1. Lược sử các phương pháp lập trình

- Lập trình tuyến tính
- Lập trình cấu trúc
- Lập trình module
- Lập trình hướng đối tượng

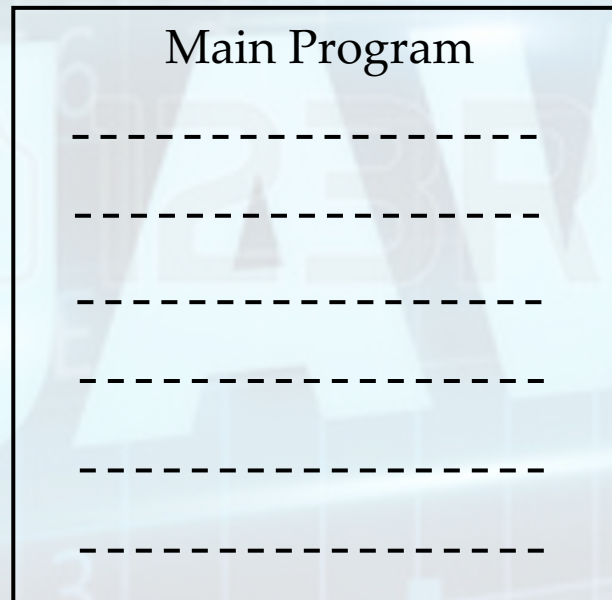
2. Đặc điểm lập trình OOP (OOP)

- Trừu tượng hoá
- Đóng gói
- Kế thừa
- Đa hình



Lập trình tuyến tính

- Chương trình thực hiện tuần tự từ đầu đến cuối.



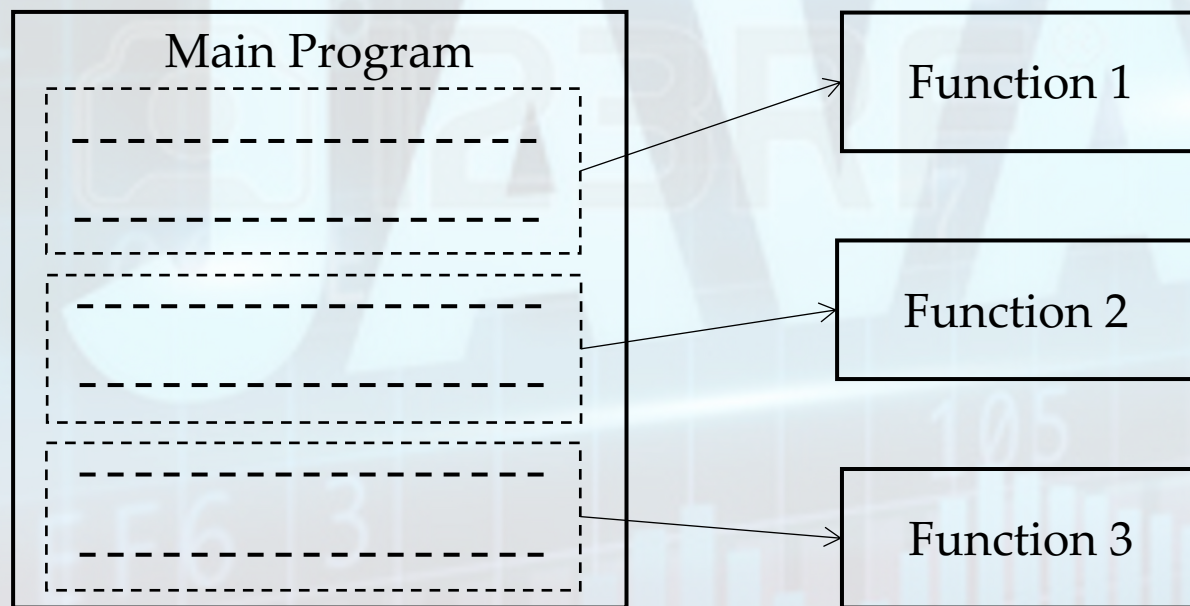


Lập trình tuyến tính

- Ưu điểm: đơn giản, dễ hiểu.
- Khuyết điểm
 - Chương trình khó đọc.
 - Khó áp dụng cho các ứng dụng lớn, phức tạp.
 - Không tái sử dụng được mã nguồn.
 - Không kiểm soát sự thay đổi dữ liệu.
 - Khó mở rộng và bảo trì.

Lập trình cấu trúc

- Lập trình cấu trúc hay lập trình thủ tục chia chương trình thành các phần nhỏ hơn gọi là các hàm hay thủ tục.





Lập trình cấu trúc

- Chú trọng các thủ tục hoặc thuật toán để giải quyết vấn đề trước, rồi mới quan tâm cấu trúc dữ liệu xử lý lưu trữ.
- Chương trình = thuật toán + cấu trúc dữ liệu

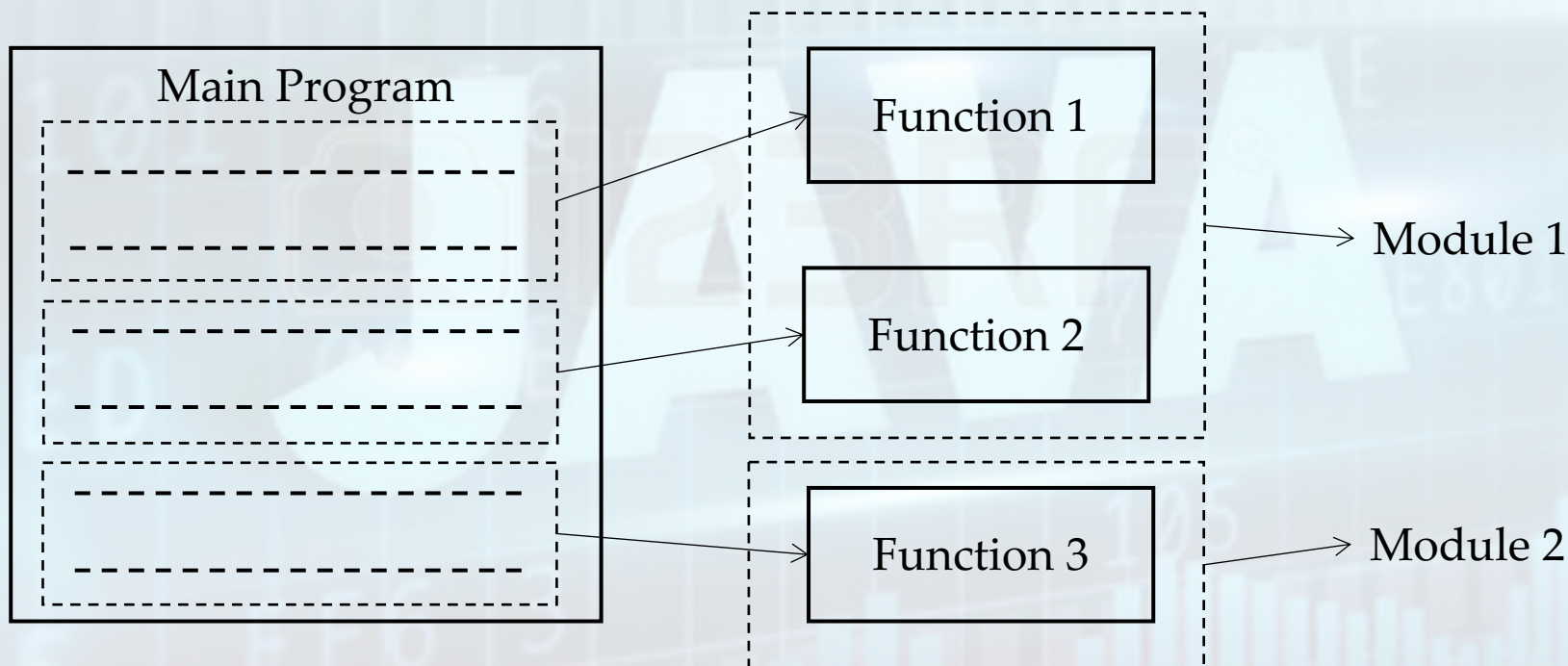


Lập trình cấu trúc

- Ưu điểm
 - Cấu trúc rõ ràng, đơn giản, dễ hiểu.
 - Dễ bảo trì và quản lý.
- Nhược điểm
 - Khó quản lý sự thay đổi dữ liệu.
 - Chương trình khó mở rộng.
 - Khả năng tái sử dụng mã nguồn hạn chế.

Lập trình module

- Lập trình module gom các thủ tục, chức năng giống nhau thành module.





Lập trình module

- Ưu điểm
 - Phù hợp phát triển ứng dụng lớn.
 - Cấu trúc chương trình rõ ràng, mạch lạc.
 - Mã nguồn dễ dàng bảo trì và phát triển.
- Khuyết điểm
 - Các khuyết điểm giống lập trình cấu trúc.



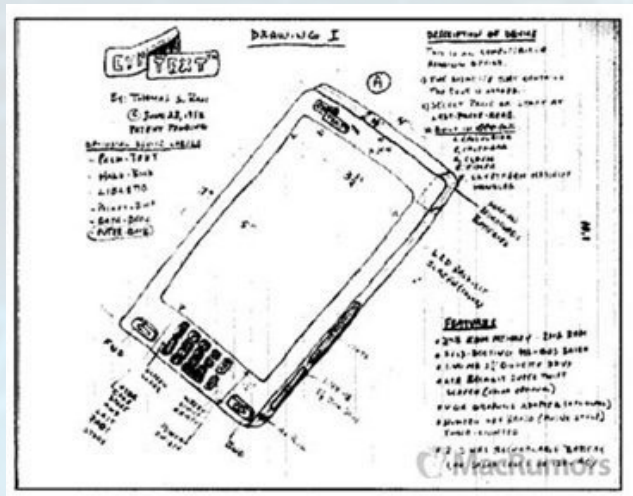
Lập trình hướng đối tượng

- Lập trình hướng đối tượng (Object-oriented Programming: OOP) là phương pháp lập trình dựa trên kiến trúc **lớp** và **đối tượng**.
- Lập trình hướng đối tượng chú trọng tổ chức dữ liệu trước, rồi mới quan tâm thuật toán giải quyết vấn đề.

Lập trình hướng đối tượng

- OOP cho phép **che giấu dữ liệu**, hạn chế tối đa truy cập từ bên ngoài.
- OOP cho phép **tái sử dụng mã nguồn** hiệu quả, tiết kiệm thời gian của lập trình viên, cũng như giúp cho việc mở rộng chương trình, bảo trì trở nên dễ dàng hơn.

Lập trình hướng đối tượng



Lớp



Đối tượng



Đối tượng



Đối tượng



Đối tượng

Nguồn ảnh: internet



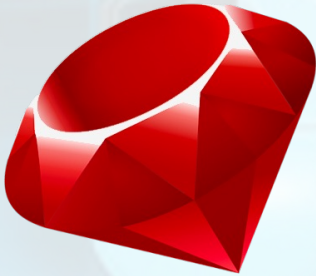
Lập trình hướng đối tượng

- Ưu điểm
 - Che giấu dữ liệu.
 - Khả năng tái sử dụng mã nguồn cao.
 - Chương trình dễ dàng mở rộng.
- Khuyết điểm
 - Phức tạp.
 - Khó theo dõi luồng dữ liệu.
 - Không thích hợp các hệ thống nhỏ.



Lập trình hướng đối tượng

- Các ngôn ngữ hỗ trợ hướng đối tượng

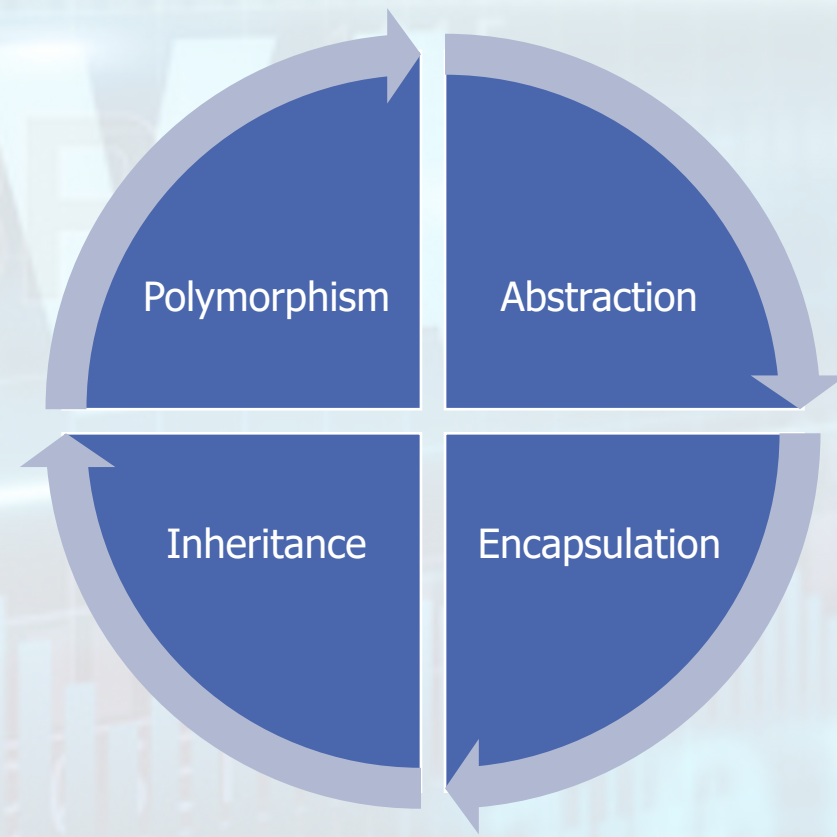


Swift



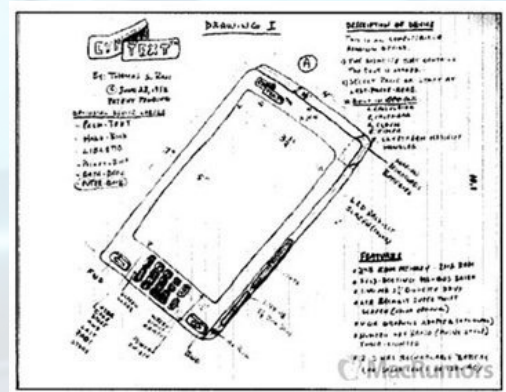
Lập trình hướng đối tượng

- Các đặc điểm của lập trình hướng đối tượng
 - Trừu tượng hoá (Abstraction)
 - Đóng gói (Encapsulation)
 - Kế thừa (Inheritance)
 - Đa hình (Polymorphism)



Trừu tượng hoá

- Trừu tượng hoá là biểu diễn một tình huống phức tạp trong thế giới thực bằng mô hình đơn giản hóa, **tập trung vào các tính chất quan trọng** của đối tượng để giải quyết một bài toán cụ thể.



Từ các thực thể giống nhau có thể trừu tượng hoá thành một lớp.



Trừu tượng hoá

- Trừu tượng hóa đối tượng theo chức năng: là quá trình mô hình hóa các phương thức của lớp dựa trên các hành động của các đối tượng.
- Trừu tượng hóa đối tượng theo dữ liệu: là quá trình mô hình hóa các thuộc tính của lớp dựa trên các thuộc tính của đối tượng.



Trừu tượng hoá

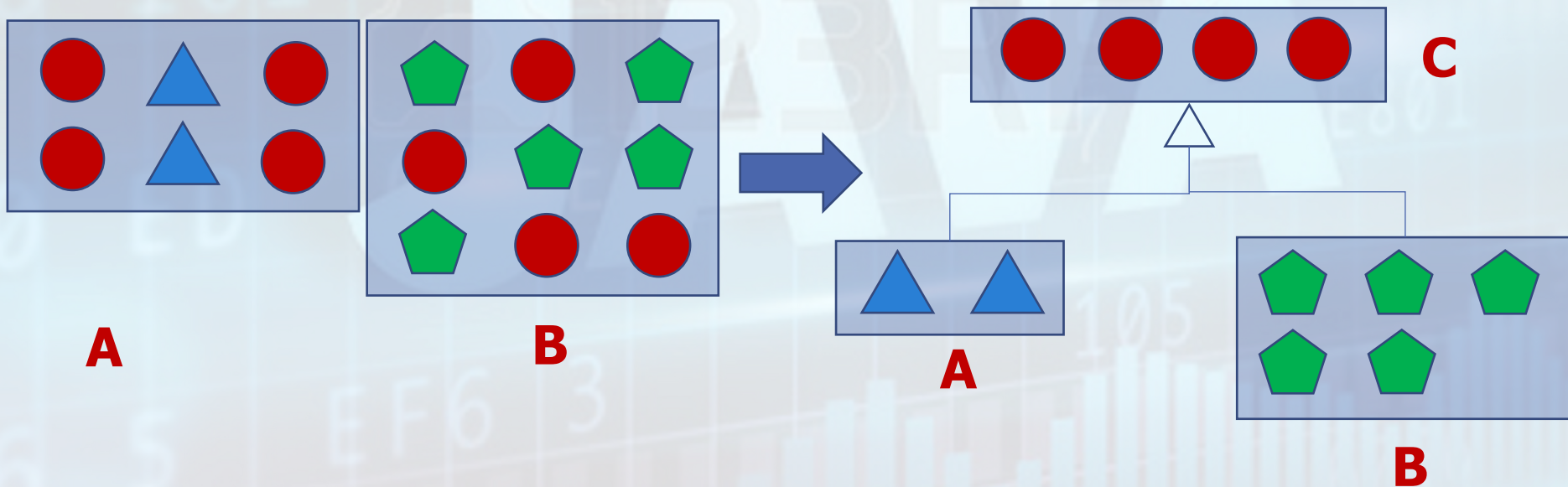
- Trong Java, trừu tượng hoá được thể hiện thông qua **lớp trừu tượng** (abstract class) và **giao diện** (interface).

JAVA

Đóng gói

- Đóng gói là cơ chế **gom dữ liệu và thao tác** trên dữ liệu thành một thể thống nhất.
- Cơ chế này giúp **che giấu thông tin** và hạn chế thao tác tự do từ bên ngoài.
- Trong Java, tính đóng gói thể hiện thông qua **phạm vi truy cập** (access modifier).

- Kế thừa (inheritance) nhằm gom các thành phần giống nhau của các lớp để tạo thành lớp mới và các lớp ban đầu kế thừa lại lớp vừa tạo.



Đa hình

- Đa hình là khả năng các đối tượng khác nhau thực hiện cùng một hành động và cho kết quả khác nhau.

Animal



Kêu

meo meo !!!

ột ột !!!

gâu gâu !!!



Đa hình

- Trong Java, tính đa hình có thể đạt được thông qua **nạp chồng** (overloading) và **ghi đè** (overriding) phương thức.

JAVA

Q&A

