



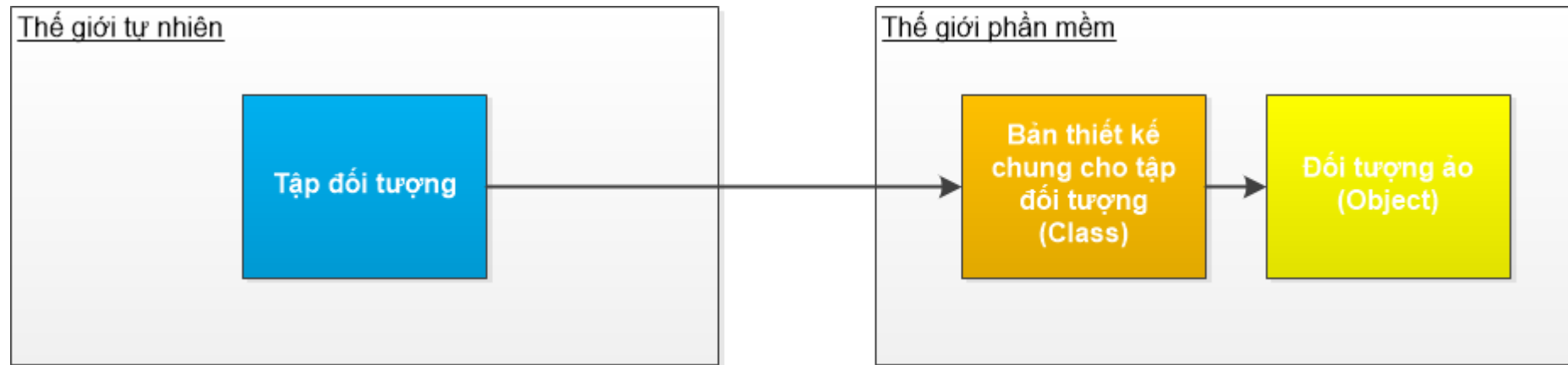
Bài 4

Lập trình hướng đối tượng

Lớp và đối tượng

- Lập trình hướng đối tượng: Đóng gói, Kế thừa, đa hình, trừu tượng
- Giới thiệu lớp và đối tượng
- Các thành phần của lớp: Instance Variable, Constructor, Method
- Bỏ từ truy cập (Access Modifier): private, protected, public, default
- Đối tượng và gọi phương thức của đối tượng
- Nạp chồng (Overloading)

- **Lập trình hướng đối tượng** – Object-Oriented Programming (**OOP**): là nghệ thuật **quan sát** các **đối tượng trong tự nhiên** rồi cố gắng nắm bắt **những hành động** cùng **đặc điểm** của chúng và **biểu diễn** dưới dạng **đối tượng ảo** trong ngôn ngữ lập trình
- **Đối tượng**:
 - Đặc điểm: đặc điểm mô tả đối tượng
 - Hành vi: hành vi (hành động) của đối tượng
- **Ưu điểm**: Dễ quản lý code, dễ mở rộng, tiết kiệm tài nguyên, bảo mật cao, tái sử dụng cao...



Encapsulation

- Che dấu những tính chất xử lý bên trong của đối tượng
- Không thể tác động trực tiếp làm thay đổi trạng thái của đối tượng mà phải thông qua các phương thức public
- Không biết luồng xử lý logic bên trong đối tượng

Inheritance

- Cho phép kế thừa lại những tính năng (đặc điểm và hành vi) của đối tượng khác
- Tránh dư thừa code

Polymorphism

- Một thông điệp được các đối tượng khác nhau hiểu và thể hiện khác nhau
- Overloading và Overriding

Abstraction

- Trừu tượng hóa định nghĩa lên những hành động, tính chất của loại đối tượng

- Field/Attribute/Instance Variable
AccessModifier **DataType** **VariableName**
- Constructor
public **ClassName**(**Parameters**)
 Initialize values for instance variables
}
- Method
AccessModifier **ReturnDataType** **MethodName**(**Parameters**)
 Block Statement;
}
- Parameter
DataType **ParameterName**
- Access Modifier: Bỏ từ truy cập xác định phạm vi truy cập
- Quy tắc đặt tên:
 - Tên biến: giống quy tắc đặt tên biến nhưng **bắt đầu là ký tự viết hoa**
 - Tên phương thức: giống tên biến

Fields/Attributes/Instance Variables
Đặc điểm chung của tập đối tượng

Constructors
Hàm tạo - Khởi tạo đối tượng

Methods - *Hành vi của tập đối tượng*

Getter/Setter methods

Methods

public: có thể truy cập ở mọi nơi

protected: có thể truy cập trong cùng package và có hỗ trợ kế thừa

private: chỉ có thể truy cập trong cùng lớp

default: có thể truy cập trong cùng package và không hỗ trợ kế thừa

Access Modifier	Trong lớp	Trong Package	Ngoài Package với lớp con kế thừa	Ngoài Package với các lớp không kế thừa
public	Có	Có	Có	Có
protected	Có	Có	Có	Không
private	Có	Không	Không	Không
default	Có	Có	Không	Không

- Khởi tạo đối tượng từ Constructor của lớp

`ClassName objectName = new ConstructorOfClass(Arguments)`

- Gọi phương thức của đối tượng: sử dụng toán tử chấm “.”

`objectName.methodName(Arguments)`

- Phân biệt tham số - Parameter và đối số - argument

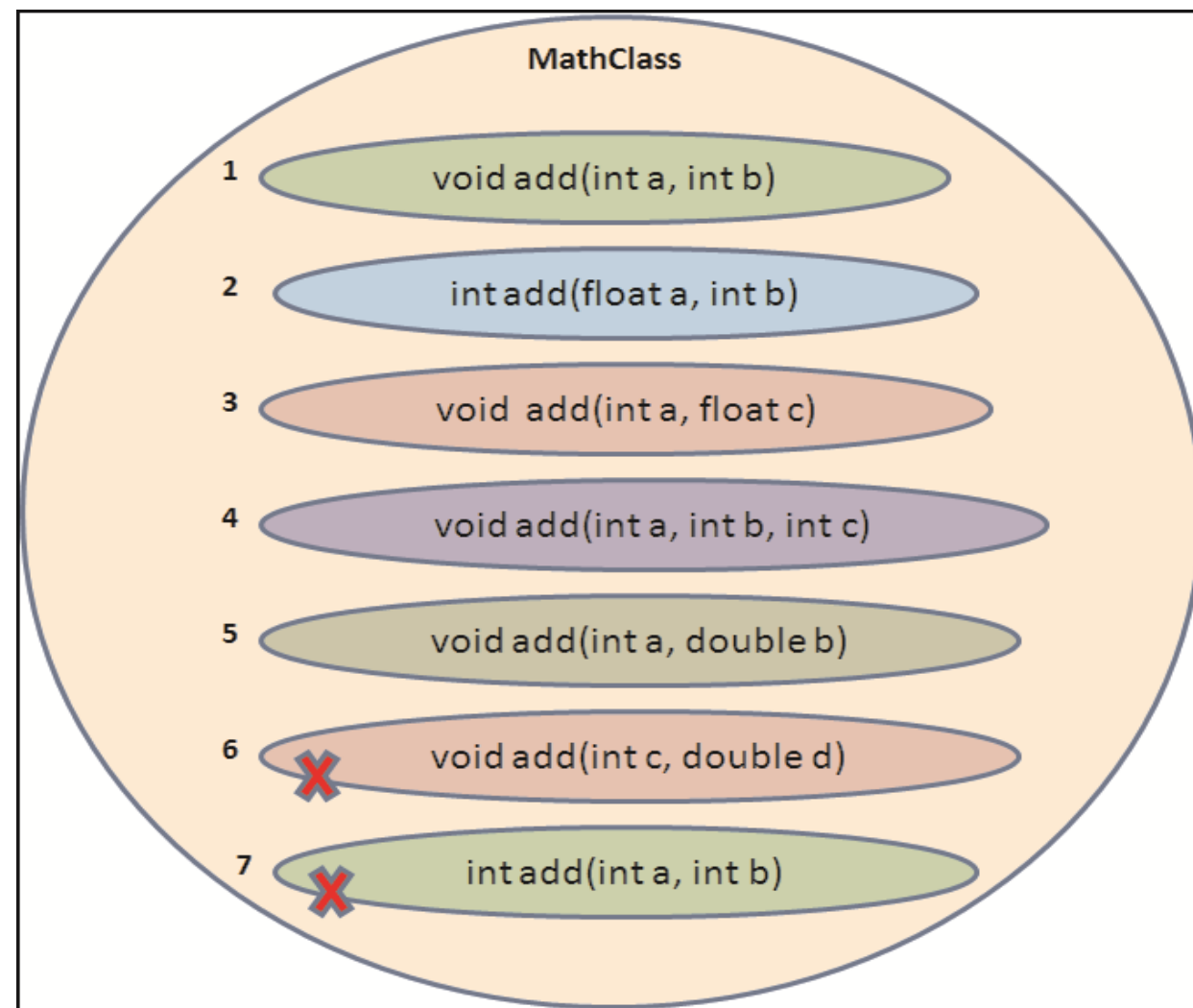
- Parameter: Tham số được định nghĩa khi khai báo phương thức của lớp
- Argument: Giá trị truyền vào khi gọi phương thức

- Argument – phân biệt truyền tham trị và tham chiếu

- Tham trị:
 - Truyền giá trị của biến (giá trị biến được lưu ở địa chỉ ô nhớ).
 - Dừng khi truyền các biến được khai báo kiểu dữ liệu nguyên thủy: byte, short, int, long, float, double, boolean, char.
- Tham chiếu:
 - Truyền địa chỉ ô nhớ của biến (ô nhớ lưu trữ địa chỉ tham khảo tham chiếu đến ô nhớ lưu trữ giá trị)
 - Dừng khi truyền các biến là các đối tượng

- Nạp chồng là các phương thức có cùng tên nhưng khác nhau:

- Số lượng tham số
- Thứ tự tham số
- Kiểu dữ liệu tham số



HỎI ĐÁP





TRẢI NGHIỆM THỰC HÀNH

HỆ THỐNG ĐÀO TẠO CNTT QUỐC TẾ BACHKHOA - APTECH



TRÂN TRỌNG CẢM ƠN!



238 Hoàng Quốc Việt, Bắc Từ Liêm, Hà Nội



0968.27.6996



tuyensinh@bachkhoa-apttech.edu.vn



www.bachkhoa-apttech.edu.vn