

ÔN TẬP LÝ THUYẾT OOP

Câu 1: Lập trình hướng đối tượng là gì?

Trả lời: Là phương pháp lập trình lấy đối tượng làm nền tảng để xây dựng thuật giải, xây dựng chương trình.

Lập trình hướng đối tượng là phương pháp lập trình dựa trên kiến trúc lớp (class) và đối tượng (object).

Câu 2: Đối tượng là gì?

Trong thế giới thực, đối tượng được hiểu như là một thực thể: người, vật hoặc một bảng dữ liệu...

- Mỗi đối tượng sẽ tồn tại trong một hệ thống và có ý nghĩa nhất định trong hệ thống.
- Đối tượng giúp biểu diễn tốt hơn thế giới thực trên máy tính
- Mỗi đối tượng bao gồm 2 thành phần: *thuộc tính và thao tác (hành động)*

Câu 3: Lớp là gì?

Các *đối tượng có các đặc tính tương tự nhau* được gom chung thành *lớp đối tượng*. Một lớp đối tượng đặc trưng bằng các thuộc tính, và các hành động (hành vi, thao tác).

- **Thuộc tính (Attribute):** Một thành phần của đối tượng, có giá trị nhất định cho mỗi đối tượng tại mỗi thời điểm trong hệ thống.
- **Thao tác (Operation):** Thể hiện hành vi của một đối tượng tác động qua lại với các đối tượng khác hoặc với chính nó.

Câu 4 : Các đặc điểm quan trọng của lập trình hướng đối tượng (OOP)

- Các lớp đối tượng – Classes
- **Trừu tượng hóa (Abstraction):** Cách nhìn *khái quát hóa* về một tập các đối tượng có chung các đặc điểm được *quan tâm* (và bỏ qua những chi tiết không cần thiết).
- **Đóng gói – Encapsulation:** *Nhóm những gì có liên quan với nhau vào làm một, để sau này có thể dùng một cái tên để gọi đến*
 - Các hàm/ thủ tục đóng gói các câu lệnh
 - Các đối tượng đóng gói dữ liệu của chúng và các thủ tục có liên quan
- **Thừa kế - Inheritance:** Là cơ chế *cho phép một lớp D có được các thuộc tính và thao tác* của lớp C, như thể các thuộc tính và thao tác đó đã được định nghĩa tại lớp D. Cho phép cài đặt nhiều quan hệ giữa các đối tượng: đặc biệt hóa (“là”) - tổng quát hóa.

- **Đa hình – Polymorphism:** Là cơ chế cho phép một tên thao tác hoặc thuộc tính có thể được định nghĩa tại nhiều lớp và có thể có nhiều cài đặt khác nhau tại mỗi lớp trong các lớp đó.

Câu 5: Phạm vi truy xuất gồm 3 từ khóa

- **Private:** Truy xuất bên trong phạm vi lớp và trong hàm bạn.
- **Protected:** Truy xuất bên trong phạm vi của lớp, trong phạm vi lớp dẫn xuất và trong hàm bạn.
- **Public:** Truy xuất trong bất kỳ nơi đâu.

Câu 6:

	Constructor	Constructor mặc định	Destructor
Định nghĩa	Dùng để khởi tạo thể hiện lớp.	Được gọi khi thể hiện được khai báo mà không có đối số nào được cung cấp.	Thường dùng để dọn dẹp cần thiết trước khi một đối tượng bị hủy.
Tên	Trùng với tên lớp.	Trùng với tên lớp.	Trùng với tên lớp nhưng có dấu ~ đằng trước
Kiểu dữ liệu trả về	Không có.	Không có.	Không có
Tham số	Tùy ý.	Không cung cấp tham số hoặc tham số mang giá trị mặc định.	Không có tham số
Số lượng	Có nhiều constructor trong lớp.	Có 1 constructor mặc định trong lớp.	Có duy nhất một destructor.

Câu 7: Kế thừa là gì?

- **Kế thừa** là một đặc điểm của ngôn ngữ dùng để biểu diễn **mối quan hệ đặc biệt hóa – tổng quát hóa giữa các lớp**. Các lớp được trừu tượng hóa và được tổ chức thành một **sơ đồ phân cấp** lớp.
- Sự kế thừa là một mức cao hơn của **trừu tượng hóa**, cung cấp một cơ chế **gom chung** các lớp có liên quan với nhau thành một mức **khái quát hóa** đặc trưng cho toàn bộ các lớp nói trên.

Câu 8: Phạm vi truy xuất hoặc **dẫn xuất** của **kế thừa**

	private	Protected	public
private			
protected	Private	Protected	Protected
public	Private	Protected	Public

- Thành phần **private** của lớp cha không được kế thừa.
- Thành phần **protected** và **public** của lớp cha kế thừa theo kiểu **protected** thì trở thành phần **protected** của lớp con.
- Thành phần **protected** và **public** của lớp cha kế thừa theo kiểu **private** thì trở thành phần **private** của lớp con.
- Thành phần **protected** của lớp cha kế thừa theo kiểu **public** thì trở thành phần **protected** của lớp con.
- Thành phần **public** của lớp cha kế thừa theo kiểu **public** thì trở thành phần **public** của lớp con.

Câu 9: Đa hình là gì ?

Là hiện tượng các đối tượng thuộc các lớp khác nhau có khả năng hiểu cùng một thông điệp theo các cách khác nhau.

Câu 10: Phương thức ảo là gì ?

- Là phương thức không có nội dung.
- Là cách **thể hiện tính đa hình** trong ngôn ngữ C++.
- Các phương thức ở lớp cơ sở có tính đa hình phải được định nghĩa là một phương thức ảo.

Câu 11: Lớp thuần ảo và lớp trừu tượng là gì?

- Lớp cơ sở trừu tượng là lớp cơ sở không có đối tượng nào thuộc chính nó.