15 Lecture - CS304

Important Subjective

1. **What is aggregation, and how does it differ from composition?**
   Answer: Aggregation is a type of association between classes in object-oriented programming where one class contains a collection of another class's objects as a member variable. Unlike composition, the contained objects can exist independently of the containing object and can be shared among multiple containing objects.

   Can a class have multiple instances of another class as member variables in aggregation?
   Answer: Yes, a class can have multiple instances of another class as member variables in aggregation.

   How does aggregation support code reuse?
   Answer: Aggregation supports code reuse by allowing for the creation of complex objects by combining simpler objects that can be shared among multiple containing objects.

   What is the purpose of using aggregation in object-oriented programming?
   Answer: The purpose of using aggregation is to create complex objects by combining simpler objects that can be shared among multiple containing objects.

   How is aggregation represented in a UML class diagram?
   Answer: Aggregation is represented in a UML class diagram with a dashed line and an arrow pointing to the contained class.

   What happens to the contained objects when the containing object is destroyed in aggregation?
   Answer: The contained objects continue to exist independently of the containing object in aggregation.

   Can the contained objects be shared among multiple containing objects in aggregation?
   Answer: Yes, the contained objects can be shared among multiple containing objects in aggregation.

   What are some real-world examples of aggregation?
   Answer: Some real-world examples of aggregation include a house's rooms and furniture, a library's books and shelves, and a computer's peripherals and components.

   How does aggregation differ from inheritance?
   Answer: Aggregation is a type of association between classes where one class contains a collection of another class's objects as a member variable, while inheritance is a mechanism that allows a subclass to inherit properties and behaviors from a parent class.

   What is the benefit of using aggregation over composition?
   Answer: The benefit of using aggregation over composition is that it allows for greater flexibility

and reusability of objects, as the contained objects can exist independently of the containing object and can be shared among multiple containing objects.