22 Lecture - CS304

Important Mcqs

1. **Inheritance in C can be implemented using:**
   a) Structures and function pointers
   b) Classes and objects
   c) Inheritance keyword
   d) None of the above
   Answer: a) Structures and function pointers

   Which of the following is not a type of inheritance?
   a) Single inheritance
   b) Multiple inheritance
   c) Hierarchical inheritance
   d) Parallel inheritance
   Answer: d) Parallel inheritance

   The derived class inherits:
   a) All the properties and methods of the base class
   b) Only the properties of the base class
   c) Only the methods of the base class
   d) None of the above
   Answer: a) All the properties and methods of the base class

   What is the syntax to define a derived structure in C?
   a) struct Derived : Base {}
   b) struct Derived extends Base {}
   c) struct Derived : public Base {}
   d) struct Derived : private Base {}
   Answer: c) struct Derived : public Base {}

   Inheritance is used to:
   a) Achieve code reusability
   b) Encapsulate data
   c) Control access to data
   d) None of the above
   Answer: a) Achieve code reusability

   Which type of inheritance allows a derived class to inherit from multiple base classes?
   a) Single inheritance
   b) Multiple inheritance
   c) Hierarchical inheritance
   d) Hybrid inheritance
   Answer: b) Multiple inheritance

   Which keyword is used to call the constructor of the base class from the derived class

constructor?
a) super
b) base
c) this
d) parent
Answer: b) base

Which type of inheritance involves creating a new class that inherits from a base class, and then creating another class that inherits from the new class?
a) Single inheritance
b) Multiple inheritance
c) Hierarchical inheritance
d) Hybrid inheritance
Answer: c) Hierarchical inheritance

Which of the following is not a benefit of inheritance?
a) Code reusability
b) Improved maintainability
c) Reduced coupling
d) Increased code complexity
Answer: d) Increased code complexity

Which of the following is an example of polymorphism?
a) Overriding a method in a derived class
b) Calling a method from the base class
c) Inheriting properties from a base class
d) None of the above
Answer: a) Overriding a method in a derived class