

30 Lecture - CS304

Important Subjective

1. **What is the advantage of using polymorphism in a payroll application?**

Answer: The advantage of using polymorphism is that it allows for flexibility and extensibility in the payroll application. By using polymorphism, new employee types can be added without having to modify the existing code.

How can you implement polymorphism in a payroll application?

Answer: Polymorphism can be implemented in a payroll application by using inheritance and virtual functions. The base class can have virtual functions that are overridden by the derived classes. This allows the application to call the appropriate function based on the type of the employee.

What is the difference between static and dynamic polymorphism?

Answer: Static polymorphism is resolved at compile-time, while dynamic polymorphism is resolved at run-time. In C++, static polymorphism is achieved through function overloading, while dynamic polymorphism is achieved through virtual functions.

How can you implement a payroll system that handles different employee types, such as salaried and hourly employees?

Answer: A payroll system that handles different employee types can be implemented using inheritance and polymorphism. A base class Employee can be created with virtual functions for calculating pay. Derived classes, such as SalariedEmployee and HourlyEmployee, can be created that inherit from the Employee class and override the pay functions.

How can polymorphism improve the maintainability of a payroll application?

Answer: Polymorphism improves the maintainability of a payroll application by making it easier to add new employee types and modify existing ones. With polymorphism, new employee types can be added by simply creating a new derived class that inherits from the Employee base class and overrides the necessary functions.

What is the advantage of using a virtual destructor in a polymorphic class hierarchy?

Answer: The advantage of using a virtual destructor in a polymorphic class hierarchy is that it ensures that the destructor of the derived class is called when an object of the derived class is destroyed through a pointer to the base class.

How can you prevent slicing when passing objects of derived classes by value to functions that take parameters of the base class type?

Answer: To prevent slicing, objects of derived classes should be passed by reference or by pointer to functions that take parameters of the base class type.

What is the purpose of a pure virtual function in an abstract base class?

Answer: A pure virtual function in an abstract base class is a function that has no implementation and must be overridden by any derived class. This ensures that any derived class is forced to provide an implementation for the function.

What is the difference between an abstract class and a concrete class?

Answer: An abstract class is a class that has at least one pure virtual function and cannot be instantiated, while a concrete class is a class that can be instantiated and does not have any pure virtual functions.

How can you implement a payroll system that handles overtime pay for hourly employees?

Answer: A payroll system that handles overtime pay for hourly employees can be implemented by adding a virtual function for calculating overtime pay to the HourlyEmployee derived class. The function can be called in the payroll calculation function to calculate the total pay for the employee.