Date : 15-04-2023 Approved by :

Assignment: 10 Sheet : 1 OF 1

Course :PROGRAMMING Batch No. :

Module :CPP Type Practical : Internal

Time : Marks :

**Single Inheritance**

**1. Basic Inheritance: Create a Simple Class Hierarchy**

**Problem Statement:** Create a simple class hierarchy using single inheritance in C++. Define two classes: Person and Employee, where Employee is derived from Person.

* The Person class should have the following attributes: name (string) and age (integer).
* The Employee class should have an additional attribute: salary (float).
* Implement a constructor for both classes to initialize the attributes, a function to display the details of each class, and a destructor to clean up.

Write the complete code to implement the hierarchy and display an employee's details.

**2. Method Overriding with Single Inheritance**

**Problem Statement:** Consider the base class Animal and the derived class Dog.

* The Animal class has a virtual function sound(), which outputs a generic message "Animal makes a sound".
* The Dog class overrides the sound() function to output "Dog barks".

Write the C++ code to demonstrate single inheritance and method overriding. Create objects of both Animal and Dog, and call their sound() function.

**3. Single Inheritance with Constructors and Destructors**

**Problem Statement:** Create a class Shape and a derived class Rectangle to represent a rectangle.

* The Shape class should have attributes width and height, initialized through a constructor.
* The Rectangle class should have a constructor that calls the Shape class constructor to initialize the width and height. It should also have a method area() to calculate and return the area of the rectangle.

Implement the constructors and destructors for both classes, and write the code to create a Rectangle object and display its area.

**Multilevel Inheritance**

**1. Multilevel Inheritance: Simple Class Hierarchy**

**Problem Statement:** Create a multilevel inheritance hierarchy with three classes: Vehicle, Car, and ElectricCar.

* The Vehicle class should have attributes make (string) and model (string), and a method displayInfo() to show these attributes.
* The Car class should be derived from Vehicle and add an attribute fuelType (string). It should also have a method displayFuelType() to display the fuel type.
* The ElectricCar class should be derived from Car and add an attribute batteryCapacity (float). It should also have a method displayBatteryCapacity() to display the battery capacity.

Write the C++ code to display the full details of an ElectricCar object, which will include information from all three classes.

**2. Multilevel Inheritance with Method Overriding**

**Problem Statement:** Design a multilevel inheritance system with the following classes: Person, Student, and GraduateStudent.

* The Person class should have attributes name (string) and age (int). Implement a method displayInfo() to display these details.
* The Student class should be derived from Person and add an additional attribute rollNumber (int). Override the displayInfo() method to include the roll number.
* The GraduateStudent class should be derived from Student and add a thesisTitle (string) attribute. Override the displayInfo() method again to show the thesis title.

Write the C++ code to display information about a GraduateStudent object, demonstrating how method overriding works in multilevel inheritance.

**3. Multilevel Inheritance with Constructors and Destructors**

**Problem Statement:** Create a class hierarchy for employees with three levels: Employee, Manager, and Executive.

* The Employee class should have attributes name (string), id (int), and salary (float), with a constructor to initialize these attributes.
* The Manager class should inherit from Employee and add an attribute department (string). Implement a constructor that calls the base class constructor to initialize the Employee attributes.
* The Executive class should inherit from Manager and add an attribute companyCar (string). Implement a constructor that calls the base class constructor to initialize both Manager and Employee attributes.

.

Date : 15-4-2023 Approved by :

Question Paper :ITC-EDU-IE-1204-S-069-1211 Sheet : 2 OF 3

Question Paper :ITC-EDU-IE-1204-S-069-1211 Sheet : 3 OF 3

|  |  |
| --- | --- |
| In this problem, implement the constructors and destructors in each class to ensure proper initialization and cleanup. Write a program to create an Executive object and display all the details about the employee. |  |

Date : Approved by :