Date : 15-04-2023 Approved by :

Assignment: 1 Sheet : 1 OF 1

Course :PROGRAMMING Batch No. :

Module :CPP Type Practical : Internal

Time : Marks :

1. Write a C++ program that demonstrates function overloading. Implement a function called add() that can take two integers, two floats, and two strings and return their sum.
2. Create a C++ program that overloads the + operator for a class called Complex to add two complex numbers (with real and imaginary parts).
3. Write a base class Shape with a virtual function area(). Derive classes Circle and Rectangle and implement their own area() function. Demonstrate runtime polymorphism by using pointers to the base class to call the area() function.
4. Implement an abstract class Employee with a pure virtual function calculateSalary(). Derive two classes FullTimeEmployee and PartTimeEmployee and implement the calculateSalary() function for each. Demonstrate polymorphism by calling the calculateSalary() function using base class pointers.
5. Create a class MathOperations with overloaded functions multiply(). Implement one version that accepts two integers, and another that accepts three integers. Provide default values for the third integer.
6. Write a C++ program to demonstrate the difference between static (compile-time) polymorphism using function overloading and dynamic (runtime) polymorphism using virtual functions.
7. Create a base class Vehicle with a constructor that initializes a string for the vehicle name. Derive classes Car and Bike from Vehicle and call the base class constructor from the derived class constructors to initialize the name. Use polymorphism to display the vehicle's name dynamically.
8. Implement a C++ program where class A has a virtual function display(). Class B and C derive from class A. Class D inherits from both B and C. Demonstrate polymorphism with class D.
9. Create a class Point with two data members x and y. Overload the << operator to print the coordinates of the point in the format (x, y).
10. Write a program where you declare a function pointer in a class Shape and assign the pointer to different derived class functions (Circle::area() and Rectangle::area()). Demonstrate polymorphism using function pointers.

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

|  |  |
| --- | --- |
| Theory Question:-  What is **polymorphism** in C++? Explain the difference between **compile-time polymorphism** and **run-time polymorphism**. Provide examples of how **function overloading** and **virtual functions** achieve these types of polymorphism. |  |

Date : Approved by :