U.S.N.										

B.M.S. College of Engineering, Bengaluru-560019

Autonomous Institute Affiliated to VTU

January / February 2025 Semester End Main Examinations

Programme: B.E. **Semester: V**
Branch: Artificial Intelligence and Machine Learning **Duration: 3 hrs.**
Course Code: 23AM5PCOOP **Max Marks: 100**
Course: Object Oriented Programming

Instructions: 1. Answer any FIVE full questions, choosing one full question from each unit.
2. Missing data, if any, may be suitably assumed.

Important Note: Completing your answers, compulsorily draw diagonal cross lines on the remaining blank pages. Revealing of identification, appeal to evaluator will be treated as malpractice.

		UNIT - I	CO	PO	Marks
1	a)	Explain the fundamental principles of Object-Oriented Programming (OOP) and how they are applied when modeling physical objects in Python.	*CO1*	*PO1*	**10**
	b)	Design a class Vehicle that takes make, model and engine_type features and a method display () to print the details. Implement parameterized and non-parameterized constructors for objects Honda and Hyundai.	*CO1*	*PO2*	**10**
		OR			
2	a)	Illustrate the concept of Abstract machines. Also detail on Interpreter working mechanism with a neat flow chart.	*CO1*	*PO1*	**10**
	b)	Define a constructor. Design a class that accepts employee name, id and salary and print the same details.	*CO1*	*PO2*	**10**
		UNIT - II			
3	a)	Design a class named MovieTicket to facilitate movie ticket bookings having following attributes: i. Movie_name ii. Show_time iii. Ticket_price iv. Number_of_tickets_available. Implement the following methods: i. Book_tickets- Allow users to book a specified number of movie tickets. Update the number_of_tickets_available attribute and return a confirmation message if the booking is successful, or an error message if there are not enough tickets available. ii. Calculate_total_cost- Calculate and return the total cost of the specified number of tickets based on the ticket_price.	***CO2***	***PO 2***	**10**

			iii. Display_movie_info: Display information about the movie, including its name, show time, ticket price, and the number of available tickets.			
		b)	Design a function called is_valid_email that takes an email address as an argument and returns True/False depending on whether it is a valid email address or not. Apply the following rules for email validation: i. Must start with a letter or a number. ii. Contain at least 1 character before the @ symbol iii. Should have at-least 1 character after the @ symbol and before the period(.) iv. Must contain at least 1 character after the last period(.). v. Maximum 256 characters	***CO2***	***PO3***	**10**
			OR			
	4	a)	Construct a Product class with the following attributes i. Product_id ii. Product_name iii. Product_price Implement the following methods: i. The Apply_discount method calculates the discounted price based on the given discount percentage. ii. The Is_expensive method checks if the product price exceeds a specified threshold. iii. The Display_product_info method is utilized to print information about the product.	***CO2***	***PO2***	**10**
		b)	i. Write a program which can map() to make a list whose elements are cube of elements in a given list. ii. If two consecutive odd numbers are both prime then they are known as twin primes. Write a program to print twin primes less than 1000.	***CO2***	***PO2***	**10**
			UNIT - III			
	5	a)	Create a hierarchical structure of classes for an employee management system. The base class is *Employee*, and there are three derived classes: *Manager, Developer, and Intern*. Each class should have attributes such as name, employee_id, salary, and additional attributes specific to the role (e.g., num_employees_managed for managers, programming_language for developers). Also implement methods for calculating bonuses based on the role and displaying employee data.	***CO3***	***PO3***	**10**
		b)	Build a shopping cart system for an e-commerce website and demonstrate with proper working code for the following design requirements: i. Create abstract classes for Cart and Product. ii. Implement derived classes for different product types. iii. Use encapsulation to manage cart operations securely.	***CO3***	***PO3***	**10**

			OR			
	6	a)	Develop a class structure for electronic devices like smartphones, laptops, and smartwatches. Use multilevel inheritance to model the relationships between a base class Device and its derived classes. Include attributes like brand, model, and battery life, and incorporate methods to perform device-specific actions.	***CO2***	***PO2***	**10**
		b)	In what scenario do multiple inheritance fails in python? Address the issues associated and provide solutions to tackle the design issue?	***CO2***	***PO2***	**10**
			UNIT - IV			
	7	a)	Demonstrate a class hierarchy for employees in a company. Include a base class Employee and derived classes Manager, Developer, and Tester. Implement polymorphic methods such as calculate_salary() for each role, considering variations in salary structures.	***CO3***	***PO2***	**10**
		b)	Provide a complete working code for the following requirements. i. The program should prompt for user input until the guessed number is correct or not (Assuming the value of the guessed number =20). ii. The program should intimate the user with a valid error if the number is too large or too small.	***CO3***	***PO2***	**10**
			OR			
	8	a)	i. Differentiate error & exception. ii. Anticipate what type of valid errors can be handled in a proper order. **try:** x = 1 / "a" y = open("file_not_exist.txt") z = "a" + None **except**(Fix Code here) as Fix Code here: print("Error:", Fix Code here) x, y, z = None, None, None print("x:", x) print("y:", y) print("z:", z)	***CO3***	***PO2***	**10**
		b)	Provide a detailed illustration of how method overloading and method overriding contribute to achieving polymorphism in object-oriented programming.	***CO3***	***PO2***	**10**
			UNIT - V			
	9	a)	Design a CustomTimer class functioning as a timer with diverse timing approaches, incorporating methods for initiation, cessation, and reset.			

			i. Implement two strategies—utilizing the time module and the threading module—for creating the timer. ii. Additionally, create a program that consistently exhibits the current time. iii. Develop the function display_current_time() to print the current time and employ a timer to execute this function every second, showcasing a continuous display.	***CO3***	***PO3***	**10**
		b)	Describe in detail the working principles of reference counting and garbage collection mechanisms in python with suitable examples.	***CO3***	***PO1***	**10**
			OR			
	10	a)	Is Global Interpreter Lock (GIL) necessary? Justify any solution if GIL is not required. Also write a python code that implements a concurrency mechanism for printing numbers (1 to 5) and alphabets (A to D) in a span of 10 seconds.	***CO3***	***PO3***	**10**
		b)	Answer the following i. Identify and justify which of the following is CPU bound task or I/O bound task 1. A program that computes new digits of Fibonacci series. 2. A program that looks through a huge data inside a video file. ii. For the above scenario do you use concurrency or parallelism? iii. Provide a python program that eliminates race conditions among two threads where one thread tries to print 5 student names and another thread tries to print the age of those 5 students. (Assume time for threads to complete their job is 10s).	***CO3***	***PO1***	**10**

****