U.S.N.										

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

Autonomous Institute Affiliated to VTU

April 2024 Semester End Main Examinations

Programme: B.E. **Semester: III**

Branch: CSE (IoT & Cybersecurity including Blockchain) **Duration: 3 hrs.**

Course Code: 23IC3PCOOP **Max Marks: 100**

Course: Object Oriented Programming with C++

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)	How does Object-Oriented Programming (OOP) enhance code organization, reusability, and maintainability? Provide examples illustrating these benefits in real-world scenarios.	*CO1*	*PO1*	**08**
	b)	Design a class called 'Book' in C++ to store details of the book such as the name and cost of a book, and include member functions 'getDetails' and 'putDetails' to respectively read and display the book details. Demonstrate how to instantiate an array of Book objects to utilize this class effectively?	*CO3*	*PO2*	**08**
	c)	With suitable examples demonstrate the usage of access specifiers in C++ and how do they control the visibility of class members?	*CO1*	*PO2*	**04**
		UNIT - II			
2	a)	With suitable C++ code demonstrate implementation of a friend function named FIND_MAX in C++, which compares the integer data of two classes, Funct1 and Funct2, and prints the maximum value found? Assume that both classes have a single integer data member.	*CO3*	*PO2*	**08**
	b)	Explain the role of constructors in C++, and how do they differ from regular member functions. With examples demonstrate the initializing objects using default constructors, parameterized constructors, and copy constructors?	*CO1*	*PO1*	**08**
	c)	Explain inline functions. What are the advantages of inline functions?	*CO1*	*PO1*	**04**
		UNIT - III			
3	a)	List the operators which cannot be overloaded. Write a program to add and subtract two complex numbers by overloading the operator + and – with suitable messages.	*CO3*	*PO3*	**10**
	b)	What is multiple inheritance? With an example program demonstrate how the ambiguity that could arise sometimes in multiple inheritance can be resolved.	*CO2*	*PO1*	**10**
		OR			

	4	a)	What is operator overloading in C++, and what are the essential rules for effectively implementing it? Illustrate these concepts with examples demonstrating operator overloading and the associated rules.	*CO2*	*PO1*	**08**
		b)	Define inheritance and importance of inheritance, also explain different forms of inheritance with examples.	*CO1*	*PO1*	**08**
		c)	Explain the visibility of base class members with different derivations.	*CO1*	*PO1*	**04**
			UNIT - IV			
	5	a)	Illustrate the difference between overloading unary and binary operator using operator overloaded function and friend function with examples.	*CO1*	*PO1*	**08**
		b)	What is implicit type conversion in C++ and how does it facilitate the conversion of smaller rank data types into higher ones illustrate with example.	*CO2*	*PO1*	**08**
		c)	With suitable example explain the Typecast operator.	*CO1*	*PO1*	**04**
			OR			
	6	a)	Design a base class Circle with member variables (radius and color) of type double, methods (getRadius(), getArea()) and constructors (Circle(radius), Circle(radius, color)). Derive subclass called Cylinder from the superclass Circle with member variable (height) of type double, public methods (getHeight(), getVolume(), getArea()) and its constructors(Cylinder(height, radius), Cylinder(height, radius,color)). Create the two instances of cylinder and print similar cylinders if the area, volume and color of cylinders are same. Demonstrate the code reuse and polymorphism properties of Object-oriented programming by inheriting the constructors and methods of the base class.	*CO3*	*PO3*	**08**
		b)	Write a C++ program to copy content of one file to another file until end of file is reached and display the copied content to the output screen.	*CO3*	*PO2*	**08**
		c)	Define pure virtual function with example.	*CO1*	*PO1*	**04**
			UNIT - V			
	7	a)	Create a class template Vector for performing the scalar product of both int type and float type	*CO3*	*PO2*	**08**
		b)	Design a container class capable of dynamically storing 500 data items. Define functions within this class to read and display the data, and provide a user program to elucidate its operational aspects during execution.	*CO3*	*PO3*	**08**
		c)	Illustrate with example application of container class.	*CO1*	*PO1*	**04**

****