Object Oriented Programming through C++

Course Code		**Year**	III	**Semester**	I
Course Category	IDE	**Branch**	CSE	**Course Type**	Theory
Credits	3	**L-T-P**	3-0-0	**Prerequisites**	Programming for Problem Solving
Continuous Internal Evaluation:	30	**Semester end evaluation**	70	**Total Marks**	100

Course Outcomes		
Upon successful completion of the course, the student will be able to		
CO1	Understand the principles of OOPs and key features of C++	**L2**
CO2	Apply object oriented concepts to developing solution for the given problem.	**L3**
CO3	Apply generic programming and STL concepts to develop effective solution for the given problem.	**L3**
CO4	Analyze the given scenario and choose appropriate OOP concept to solve the given problem.	**L4**

Contribution of Course Outcomes towards achievement of Program Outcomes & Strength of correlations (3:Substantial, 2: Moderate, 1:Slight)

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	3												1	1
CO2	3								3	3			2	2
CO3	3								3	3			2	2
CO4		2							3	3			2	2

Syllabus		Mapped CO
Unit No.	**Contents**	
I	**Introduction–** Need for OOP Paradigm, Procedural Programming vs Object Oriented Programming, Object Oriented Concepts, CIN & COUT **Functions–** Main function, Function Prototyping, Inline Functions, Reference Variables, Call by Reference, Defaults Arguments, Function Overloading, Library Functions, C++ Manipulators, Type Casting.	**CO1, CO2**
II	**Class–** Difference between C structure and class, specifying a class, Defining Member Functions, Scope Resolution Operator, Memory Allocation for Objects, Array within a Class, Array of Objects, Constructors and Destructors, Static Members, Object as Function Arguments, Returning Objects, Friend Functions, This Pointer.	**CO1, CO2**
III	**Operator Overloading–** Defining Operator Overloading, Overloading Unary and Binary Operator. **Inheritance–** Base class, Derived Class, Visibility Modes, Types of Inheritance, Overriding.	**CO1, CO2**
IV	**Virtual Functions–** Containership, Virtual Function, Pure Virtual Functions, Abstract Class, Pointer to Derived Class. **Exception Handling–** Principles of Exception Handling, Throwing Exceptions, Catching Exceptions, Defining New Exceptions.	**CO1, CO2**
V	**Generic Programming–** Templates, Definition of class Templates, Function Templates, Over Loading of Template Function. **Standard Template Library–** Introduction to STL, Algorithms, Sequence Containers, Iterators, Associative Containers.	**CO1, CO3, CO4**

Learning Resources
Text Books
1. Object-Oriented Programming in C++, Robert Lafore, 4th Edition, 2002, SAMS. 2. Object-Oriented Programming with C++, E Balagurusamy, 8th Edition, 2020, Mc Graw Hill.
Reference Books
1. The C++ Programming Language, Bjarne Stroustup, 4th Edition, 2013, Addison-Wesley. 2. Object-Oriented Programming Using C++ Paperback, Joyce Farrell, 4th Edition, 2013, Cengage.

e- Resources & other digital material
1. https://www.learncpp.com/ 2. https://onlinecourses.nptel.ac.in/noc21_cs02/preview 3. https://www.educative.io/courses/learn-object-oriented-programming-in-cpp 4. https://www.youtube.com/watch?v=wN0x9eZLix4 (Learn Object Oriented Programming in C++, Beau Carnes, February 2021)