

OBJECT ORIENTED PROGRAMMING WITH C++

SUBJECT CODE: IT 306

Teaching Scheme (Credits and Hours)

Teaching scheme				Total Credit	Evaluation Scheme					
L	T	P	Total		Theory		Mid Sem Exam	CIA	Pract.	Total
Hrs	Hrs	Hrs	Hrs		Hrs	Marks	Marks	Marks	Marks	Marks
03	00	02	05	04	03	70	30	20	30	150

Learning Objectives:

The educational Objectives of this Course are:

Provide flexible and powerful abstraction

Allow programmers to think in terms of the structure of the problem rather than in terms of the structure of the computer.

Decompose the problem into a set of objects

Objects interact with each other to solve the problem

create new type of objects to model elements from the problem space

Outline of the Course:

Sr. No	Title of the Unit	Minimum Hours
1	Principles of Objective Oriented Programming	05
2	Token Expressions & Control Structures	05
3	Functions in C++, Classes & Objects.	10
4	Constructors & Destructors, Operator Overloading, Inheritance	10
5	Pointers, Virtual Functions & Polymorphism, Working with Files, Exception handling	09
6	An Object Oriented Approach in Real Life Problems	06

Total hours (Theory): 45

Total hours (Lab): 30

Total hours: 75

Detailed Syllabus

Sr. No	Topic	Lecture Hours	Weight age (%)
1	Principles of Objective Oriented Programming Object Oriented Programming Paradigm, Basic Concepts of Object Oriented Programming, Benefits of Object Oriented Programming, Object Oriented Languages, Applications of Object Oriented Programming, Beginning with C++.	5	12
2	Token Expressions & Control Structures Tokens, Keywords, Identifiers and Constants, Data Types, Type Compatibility, Variables, Operators in C++, Implicit Conversions, Operator Overloading, Operator Precedence, Control Structures.	5	18
3	Functions in C++, Classes & Objects. The Main Function, Function Prototyping, Call by Reference, Return by Reference, Inline Functions, Function Overloading, Friend and Virtual Functions. Specifying a class, Member Functions, Arrays within a class, Static Member Functions, Arrays of Objects, Friendly Functions.	10	18
4	Constructors & Destructors, Operator Overloading, Inheritance Constructors, Parameterized Constructors, Copy Constructors, Dynamic Constructors, Destructors, Defining Operator Overloading, Overloading Operators, Rules for Overloading Operators, Type Conversions	10	18
5	Pointers, Virtual Functions & Polymorphism, Working with Files, Exception handling Pointers, Pointers to Objects, this pointer, Pointer to Derived Classes, Virtual Functions, Classes for File Stream Operations, Opening and Closing a File, File Modes, File Pointers, Input Output Operations, Updating a File.	9	18
6	An Object Oriented Approach in Real Life Problems Object Orientation O Development O Themes, Modelling, Abstraction Models.	6	16
	TOTAL	45	100

Instructional Method and Pedagogy:

- At the start of course, the course delivery pattern, prerequisite of the subject will be discussed.
- Lectures will be conducted with the aid of multi-media projector, black board, OHP etc.
- Attendance is compulsory in lecture and laboratory which carries 10 marks in overall evaluation.
- One internal exam will be conducted as a part of internal theory evaluation.
- Assignments based on the course content will be given to the students for each unit and will be evaluated at regular interval evaluation.
- Surprise tests/Quizzes/Seminar/tutorial will be conducted having a share of five marks in the overall internal evaluation.

- The course includes a laboratory, where students have an opportunity to build an appreciation for the concepts being taught in lectures.
- Experiments shall be performed in the laboratory related to course contents.

Students Learning Outcome:

On successful completion of the course, the student will:

- Master the fundamental principles of OO programming,
- Master key principles in OO analysis, design, and development.
- Be familiar with the application of the Unified Modeling Language (UML) towards analysis and design.
- Master common patterns in OO design and implement them,
- Be familiar with alternative development processes,
- Be familiar with group/team projects and presentations.
- Be exposed to technical writing and oral presentations.

Reference Books:

1. Object Oriented Design by Rumbaugh (Pearson publication)
2. Object-oriented programming in Turbo C++ By Robert Lafore, Galgotia Publication.
3. Object-oriented programming with C++ by E.Balagurusamy, 2nd Edition, TMH.

List of experiments:

Sr. No	Name of Experiment
1	WAP to find the sum of two numbers using function.
2	WAP to find Simple Interest and Compound Interest.
3	WAP to demonstrate the working of following Loops: While, Do While, For, If-Else, switch
4	WAP to find greatest of three numbers.
5	WAP to check whether a number is even or odd.
6	WAP to check whether a year is leap year or not.
7	WAP to add and subtract two matrices.
8	WAP to display elements of an array.
9	WAP to calculate Sum and Average of an array.
10	WAP to sort elements of an array using Bubble sort.
11	WAP to calculate Factorial of a number.
12	WAP to check whether a given number is Prime or not.
13	WAP to generate Fibonacci series.
14	WAP to show function Overloading.
15	WAP to create a class and access member function of a class
16	Write a program to show Constructor and Destructor in a class
17	Write a program to convert the temperature in Fahrenheit to Celsius and vice-a-verse
18	Write a program to show the concept of Single inheritance in classes.