



Indian Institute of Information Technology, Lucknow

भारतीय सूचना प्रौद्योगिकी संस्थान, लखनऊ

Department of Information Technology

Semester: II

Course Code:

Course Name: Object Oriented Methodology

Credits	L	T	P	Section (Group)
4	3	0	1	B.Tech. (CS/IT/CS-AI)

### Course Module Details

Objective(s)	<p>The objective of this course is to introduce the students to the concepts of Object-Oriented Methodology and Design. The course covers basic concepts such as Inheritance, Abstraction and Polymorphism. In addition, it introduces students to the design process to represent the structural and behavioural aspects of a system. The students will also learn the basics of the Universal Modelling Language for documenting the design artefacts. To implement the concepts at a concrete level, we will use programming in C++ as a tool.</p>
Covered Topics	<p><b>Week 0:</b></p> <ul style="list-style-type: none"><li>• <b>Revision of C:</b> Operators, Data Types, Conditional Statements, Iterative Statements, Structures, Functions</li></ul> <p><b>Week 1:</b></p> <ul style="list-style-type: none"><li>• <b>Foundations:</b> An overview of the Software Development Lifecycle, The Waterfall Model, The Design Phase of the Waterfall Model, Iterative vs Declarative Paradigm, Levels of Abstractions</li><li>• <b>Basics of Object-Oriented Ecosystem:</b> The idea of Object-Oriented Programming, Difference between Structured and Object-Oriented Programming, Definition of basic terms - Class, Object, Attributes, Operations and Methods</li></ul>

#### Week 2:

- **Object-Oriented Modeling Approach:** Designing a System using Object-Oriented Concepts, Three types of Object-Oriented Models - Class Diagrams, State Diagrams and Interaction Diagrams
- **Introduction to C++:** Difference between C and C++, Installing and using g++, Creating and Running the “Hello World” program

#### Week 3:

- **Inheritance and Data Hiding:** The idea of Inheritance, Single Inheritance, The Open-Closed Principle
- **Inheritance in C++ - Part 1:** Creating Classes in C++, Difference between `private`, `public` and `protected` members, friend functions

#### Week 4:

- **Multiple Inheritance:** The idea of Multiple Inheritance, The Inheritance Hierarchy, Challenges of Multiple Inheritance
- **Inheritance in C++ - Part 2:** Multiple Inheritance in C++, The use of scope resolution operator

#### Week 5:

- **Polymorphism:** The idea of Polymorphism, Compile-time vs Runtime Polymorphism
- **Overloading in C++:** Function Overloading, Operator Overloading
- **Overriding in C++:** Function Overriding, Creating Virtual Functions

#### Week 6:

- **Introduction to UML:** Overview of UML, Structural vs Behavioural Modelling, Types of UML Diagrams,
- **File Handling in C++:** The concept of Streams, Reading and Writing Data to and from a File
- **Exception Handling in C++:** The concept of Exception Handling, The try-catch mechanism, The `throws` keyword

#### Week 7:

- **Class Diagrams:** Analysing the structure of real-world Objects, Capturing Structural Information with Class Diagrams
- **Building Applications with C++ - Part 1:** Introduction to Multi-threading, Using POSIX threads in C/C++

	<p><b>Week 8:</b></p> <ul style="list-style-type: none"> <li>• <b>State Diagrams:</b> States and Events, Capturing changes in the state of Objects with State Diagrams</li> <li>• <b>Building Applications with C++ - Part 2:</b> Introduction to Networking, Creating and Communicating over POSIX Sockets in C/C++</li> </ul> <p><b>Week 9:</b></p> <ul style="list-style-type: none"> <li>• <b>Interaction Diagrams:</b> Expressing interactions between Objects, Sequence Diagrams, Activity Diagrams, Use Case Diagrams</li> <li>• <b>Building Applications with C++ - Part 3:</b> Building GUI Applications with Qt</li> </ul> <p><b>Week 10:</b></p> <ul style="list-style-type: none"> <li>• <b>Deployment Diagrams:</b> Allocating software components on hardware using Deployment Diagrams</li> <li>• <b>Building Applications with C++ - Part 4:</b> Interprocess Communication in C/C++, Creating and using Shared Memories, The <code>fork()</code> system call</li> </ul> <p><b>Week 11:</b></p> <ul style="list-style-type: none"> <li>• <b>Term Project Discussions - Part 1:</b> Picking a suitable Term Project, Creating Class Diagrams for Projects</li> </ul> <p><b>Week 12:</b></p> <ul style="list-style-type: none"> <li>• <b>Term Project Discussions - Part 2:</b> Creating State, Interaction and Deployment Diagrams for Projects</li> </ul>
<b>Pre-Requisites</b>	Introduction to Programming (I Semester)

In addition, one week before the End-sem exam will be used for project evaluation.

#### Contact Details:

Saurabh Srivastava,  
Department of Information Technology,  
Indian Institute of Information Technology,  
Email: [saurabh@iiitl.ac.in](mailto:saurabh@iiitl.ac.in)

## Reference material

- [1] Michael R Blaha and James R Rumbaugh. *Object-Oriented Modeling and Design with UML*. Pearson Education India, 2005.
- [2] David A Curry. *Using C on the UNIX System*. O'Reilly, 1989.
- [3] Ashok Kamthane. *Object-Oriented Programming with ANSI and Turbo C++*. Pearson Education India, 2003.
- [4] Bjarne Stroustrup. *The C++ Programming Language, Third Edition*. Pearson Education India, 2000.