Integrated MCA I Semester 2

Subject Name: Object Oriented Programming using C++ **Subject Code:** IMCA13201

Type of course: Professional Core Course

Prerequisite (if any):
- C programming

List of Courses where this course will be prerequisite:
- Java
- .NET
- Android

Rationale:OOP has become a fundamental part of software development. OOP facilitates reuse of code, flexibility and effective problem solving. This course introduces standard tools and techniques for software development, using an object oriented approach.

Teaching and Examination Scheme:

TEACHING SCHEME				Theory Marks		Practical Marks		Total
L	T	P	C	TEE	CAT	TEP	CAP	
4	0	0	4	60	40	-	-	100

CAT: Continuous Assessment Theory comprised of CA1 and CA2 **CA1:** Continuous Assessment (assignments/projects/open book tests/closed book tests **CA2:** Sincerity in attending classes/class tests/ timely submissions of assignments/self-learning attitude/solving advanced problems **TEE:** Term End Examination **TEP:** Term End Practical Exam (Performance and viva on practical skills learned in course) **CAP:** Regular submission of Lab work/Quality of work submitted/Active participation in lab sessions/viva on practical skills learned in courses.

Content:

Sr. No.	Content	Teaching Hrs.	Module Weightage
1	**Introduction to Object Oriented Programming** Procedural Programming vs. Object Oriented Programming, Principles of Object Oriented Programming, Advantages of Object Oriented Programming, Applications of Object Oriented Programming	05	8%
2	**Introduction to C++ Programming language** Creating your First C++ Program - Compiling and Running the Program, Keywords, Data Types, Variable Declarations, Operators, Expressions and Control Structures, Functions in C++, Function Overloading, Friend and Virtual functions	12	20%
3	**Object Oriented Programming in C++** Class and Objects, Access Specifiers, Various Types of Members of a Class, Arrays of Objects, Objects as Function Parameters and Returning Objects, Constructors, Types of Constructors, Dynamic Initialization of Objects, Constructing Arrays, Destructors, Operator Overloading and Type Conversions	15	26%
4	**Inheritance** Basics of Inheritance, Types of Inheritance, Virtual Base Classes, Abstract Classes, Constructors in Derived Classes, Nesting of Classes	08	13%
5	**Pointers and Polymorphism** Introduction and Various Usages of Pointers, Pointers and Inheritance, Types of Polymorphism, Virtual Functions, Pure Virtual Functions, Virtual Constructors and Destructors	10	16%
6	**Templates and Exception Handling** Templates for Classes, Functions and Member Functions Introduction to Standard Template Library (STL), Components of STL Introduction to Exception Handling, Exception Handling Mechanism, Throwing and Catching Exceptions, Exceptions in Constructors and Destructors, Exceptions in Operator Overloaded Functions	10	17%

Suggested Specification table with Marks (Theory):

Distribution of Theory Marks					
R Level	**U Level**	**A Level**	**N Level**	**E Level**	**C Level**
25	30	10	15	10	10

Legends: R: Remembrance; U: Understanding; A: Application, N: Analyze and E: Evaluate C: Create and above Levels (Revised Bloom's Taxonomy)

Note: This specification table shall be treated as a general guideline for students and teachers. The actual distribution of marks in the question paper may vary slightly from above table.

Reference Books:

Sr. no.	Title of book /article	Author(s)	Publisher and details like ISBN	Year of publication	Publication Edition
1	Object Oriented Programming with C++	E Balagurusamy	McGraw Hill Education, ISBN: 978-1-25-902993-6	2013	6th edition
2	C: How to Program with an introduction to C++	Paul Deitel Harvey Deitel	Pearson, ISBN: 978-93-530-6282-8	2018	8th edition
3	Teach Yourself C++	Herbert Schildt	Tata McGraw Hill ISBN: 0-07-463333-3	2007	3rd edition
4	Object Oriented Programming in C++	Robert Lafore	Galgotia publication pvt ltd. Pearson Education, ISBN: 978-81-317-2282-4	2022	4th edition

Course Outcomes:

Sr. No.	CO Statement After learning this subject, students will be able to	Marks % weightage
CO-1	Understand the concepts of object oriented programming.	8%
CO-2	Learn the concepts of programming in C++ programming language.	20%
CO-3	Learn how to write programs in C++ using object oriented programming methodology.	26%
CO-4	Understand the concept of inheritance and its various types.	13%
CO-5	Understand types of polymorphism, use of pointers in polymorphism.	16%
CO-6	Learn the usage of templates, STL and exception handling mechanism.	17%

Mapping with POs:

	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO	PO
CO-1	2	2	1	1	0	1	0	2	0	0	0	0	0
CO-2	1	0	0	0	1	1	0	1	0	0	0	0	0
CO-3	2	2	1	1	0	1	0	2	0	0	0	0	0
CO-4	2	2	1	1	0	1	0	2	0	0	0	0	0
CO-5	2	2	1	1	0	1	0	2	0	0	0	0	0
CO-6	0	0	0	1	0	0	0	0	0	0	0	0	0
Rationale*													

Rationale*: Explaining why it is matching this particular program outcomes