BVM ENGINEERING COLLEGE [AN AUTONOMOUS INSTITUTION]

2CP02: OBJECT ORIENTED PROGRAMMING WITH C++
CREDITS - 4 (LTP:3,0,1)

Course Objective:
To impart the knowledge about principles of Object Oriented Programming paradigm using C++.

Teaching and Assessment Scheme:

Teaching Scheme (Hours per Week)			Credits	Assessment Scheme				
				Theory		Practical		Total Marks
L	T	P	C	ESE	CE	ESE	CE	
3	0	2	4	60	40	20	30	150

Course Contents:

Unit No.	Topics	Teaching Hours
1	**Basics:** Introduction OOP; Procedural Vs. Object Oriented Programming; Principles of OOP; Benefits and Applications of OOP; Program Structure; Namespace; Identifiers; Variables; Constants; Operators; Typecasting; Control Structures.	03
2	**Objects and Classes:** Basics of Object and Class; Private and Public members; Static Data Member and Member Function; Constructors and their types; Destructors; Type Conversion; New and Delete operators. Arrays of Objects; Reference variables.	06
3	**Functions & Overloading:** Simple functions; Call and Return by Reference; Inline functions; Macro Vs. Inline functions; Overloading of functions; Operator overloading; Default arguments; Friend functions.	05
4	**Inheritance and Dynamic Polymorphism:** Concept of Inheritance; Types of Inheritance: Single, Multiple, Multilevel, Hierarchical, Hybrid; Protected members; Method Overriding; Virtual Base Class; Pointers and Objects; 'this' pointer; Virtual and Pure Virtual functions; Implementing Dynamic Polymorphism.	06
5	**Exception Handling:** Review of Traditional Error Handling; Basics of Exception Handling; Exception Handling Mechanism; Throwing and Catching Mechanisms; Rethrowing an Exception; Specifying Exceptions.	05
6	**I/O and File Management:** Concept of Streams; 'cin' and 'cout'; Overloading of Inserter and Extractor operators; C++ Stream Classes; Unformatted and Formatted	05

Unit No.	Topics	Teaching Hours
	I/O; Manipulators; File stream and C++ classes; File Management functions; File modes; Binary and Random Files.	
7	**Templates, STL & RTTI:** Introduction; Need and Use of Templates; Function templates and Class templates; STL: Introduction, Containers, Algorithms and Iterators; RTTI: Introduction.	05
	Total	**45**

List of References:

1. E Balagurusamy, "*Object Oriented Programming with C++*", TMH (E-book available on the BVM Intranet).
2. Herbert Schlitz, "*The Compete Reference C++*", TMH.
3. Deitel, "*C++: How to Program*", PHI.
4. Ashok Kamthane, "*Object Oriented Programming with ANSI and Turbo C++*", Pearson.
5. Debasish Jana,"*C++ and Object Oriented Programming Paradigm*", PHI.
6. Saurav Sahay, "*Object Oriented Programming with C++*", Oxford.
7. Scott Meyers, "*Effective C++*", Addison-Wesley
8. Scott Meyers, "*More Effective C++*", Addison-Wesley

Course Outcomes (COs):

After successful completion of the course, the students will be able to…

1. Differentiate between Object Oriented Programming and Procedural Programming Paradigms.
2. Illustrate features of Object Oriented Programming like Encapsulation, Inheritance, Polymorphism, etc. using C++.
3. Design a solution of a given problem using Object Oriented Programming concepts.
4. Implement the solution of a given problem using C++.
5. Incorporate features like Exception handling and Templates in the solution.
6. Enhance Logical Reasoning and Programming skills.