COURSE TITLE   :   **OBJECT ORIENTED PROGRAMMING THROUGH C++**

COURSE CODE   :   3134

COURSE CATEGORY   :   B

PERIODS/WEEK   :   5

PERIODS/SEMESTER   :   75

CREDITS   :   5

TIME SCHEDULE

MODULE	TOPICS	PERIODS
1	**Basic Programming Concepts in C++**	21
2	**Concepts of Object Oriented Programming**	18
3	**Overloading of Operators and Inheritance**	18
4	**Advanced uses of C++**	18

Course General Outcomes:

Sl.	G.O	On completion of this course the student will be able :
1	1	To comprehend the basic programming concepts in C++
2	1	To understand the Concepts of Object Oriented Programming
3	1	To understand Overloading and Inheritance.
4	1	Advanced concepts of C++

Specific Outcomes:

MODULE I   Basic Programming Concepts in C++

1.1   To comprehend the basic programming concepts in C++

   1.1.1   Explain the features of C++
   1.1.2   Explain data types and expressions in c++
   1.1.3   Explain different control structures in c++
   1.1.4   Explain Arrays as Homogeneous Aggregates
   1.1.5   Explain Structures as Heterogeneous Aggregates
   1.1.6   Describe the concepts Memory Management
   1.1.7   Explain input and output with disk files.

MODULE II Concepts of Object Oriented Programming

2.1   To understand the Concepts of Object Oriented Programming
   2.1.1   State the basic concepts of Object Oriented Programming
   2.1.2   Describe parameter passing in C++ functions.
   2.1.3   Explain the functions as modularization tool.
   2.1.4   Describe Object-Oriented Programming with Functions
   2.1.5   Describe Data Encapsulation and Information Hiding.
   2.1.6   Explain classes, objects and methods
   2.1.7   Illustrate constructors with examples
   2.1.8   Describe the concepts of overloading

MODULE III Overloading of Operators and Inheritance

3.1 To understand Overloading and Inheritance.
   3.1.1   Explain Overloading of Operators
   3.1.2   Describe overloading for nonnumeric classes.
   3.1.3   Describe different types of Inheritances.
   3.1.4   Explain Visibility controls
   3.1.5   Describe class objects as data members.
   3.1.6   Explain base and derived class.
   3.1.7   Explain Constructors and Destructors for Derived Classes
   3.1.8   Discuss inheritance vs composition.

MODULE IVI Advanced concepts of C++

4.1 To understand Virtual functions and Templates
   4.1.1   Explain Virtual functions
   4.1.2   Explain conversion between nonrelated classes.
   4.1.3   Explain the usage of several base classes.
   4.1.4   Discuss subscript and function call operators.
   4.1.5   Discuss input output operators.
   4.1.6   Explain syntax of template class
   4.1.7   Discuss Exception handling mechanism in C++
   4.1.8   Describe type cast operators

<div align="center">**CONTENT DETAILS**</div>

Module I: A Brief Overview of C++

The Basic Program Structure-Preprocessor Directives-Comments-Declarations and Definitions-Statements and Expressions-Functions and Function Calls-Classes-Data and Expressions-Values and Their Types-Integral Types-Floating Point Types-Control Flow-Statements and Expressions-Conditional Statements-Iteration-C++ Jump Statements-Defined Data Types-Arrays as Homogeneous Aggregates-Structures as Heterogeneous Aggregates-Unions, Enumerations, and Bit Fields-Memory Management: Storage Classes-Using Heap- Input and Output with Disk Files.

Module II: Object-oriented programming with C++

Programming With C++ Functions-C++ Functions as Modularization Tools-Argument Promotions and Conversions-Parameter Passing in C++-Inline Functions-Parameters with Default Values-Function Name Overloading-Object-Oriented Programming with Functions-Cohesion-Coupling-Data Encapsulation-Information Hiding.

C++ Class-Basic Class Syntax-Binding Together Data and Operations-Elimination of Name Conflicts-Implementing Member Functions Outside of Class-Defining Class Objects of Different Storage Classes-Controlling Access to Class Members-Initialization of Object Instances-Constructors as Member Functions-Default Constructors-Copy Constructors-Conversion Constructors-Destructors-Timing of Constructor and Destructor Invocations-Class Scope and the Overriding of Names in Nested Scopes-Memory Management with Operators and Function Calls-Using Returned Objects in Client Code-Static Class Members.

Module III Overloading of Operators and Inheritance.

Operator Functions: Overloading of Operators-Limitations on Operator Overloading-Overloaded Operators as Class Members-Friend Functions-Operator Overloading for Nonnumeric Classes-Overloading the Assignment Operator.

Aggregation and Inheritance: Using Class Objects as Data Members-Initialization of Composite Objects Data Members with Special Properties-Container Classes-Similar Classes: Inheritance-Accessing Base and Derived Class Services-Accessing Base Components of a Derived Class Object-Scope Rules and Name Resolution Under Inheritance-Constructors and Destructors for Derived Classes-Inheritance and Composition.

Module IV:Advanced uses of C++

Virtual Functions -Conversions Between Nonrelated Classes-Conversions Between Classes Related Through Inheritance-Multiple Inheritance: Several Base Classes-Unary Operators-Subscript and Function Call Operators-Input/Output Operators.

Templates: Syntax of Template Class Definition-Template Classes with Several Parameters-Relations Between Instantiations of Template Classes-Template Specializations-Template Functions-Programming with Exceptions--Syntax of C++ Exceptions-Exceptions with Class Objects-Type Cast Operators

Text Book(s)

1. Core C++ A Software Engineering Approach-Victor Shtern-Publisher: Prentice Hall PTR-First Edition July 24, 2000
2. The Complete Reference C++, Herbert Schildt, Tata McGraw Hill Publication, Fourth Edition**,**

References

1. E.Balaguruswamy, *Object Oriented Programming in C++,* Mc Graw Hill,Sixth Edition
2. Venugopal, Rajkumar, Ravishankar, *Mastering C++*, Mc Graw Hill,1999
3. Stroustrup, Bjarne, *The C++ Programming Language* , Addison Wesley, Fourth Edition
4. Robert Lafore, *Object Oriented Programming in C++*, Galgotia 1991