

## Object Oriented Concepts using C++

Semester – IV	Subject Code: BS41601	Lectures: 60
---------------	-----------------------	--------------

**Objectives:**

The syllabus aims in equipping students,

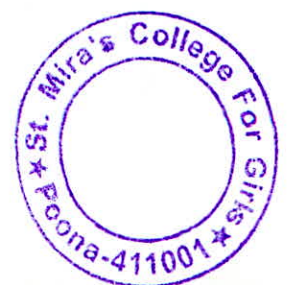
- To acquire an understanding of basic object oriented concepts and the issues involved in effective class design
- To write C++ programs that use object oriented concepts such as information hiding, inheritance, polymorphism etc.

<b>Unit 1: Object Oriented Concepts</b>	<b>8</b>
<p><b>1. Introduction to object oriented concepts</b></p> <ul style="list-style-type: none"> <li>• Characteristics of object oriented language (object, class, encapsulation ,inheritance and polymorphism)</li> <li>• Advantages and Applications of OOP [ Ref. book 1- Chapter 1] [ Ref. book 2- Chapter 1]</li> </ul> <p><b>2. Introduction to C++</b></p> <ul style="list-style-type: none"> <li>• Data types, new operators in C++ (scope resolution, member dereferencing ) and keywords, using namespace concept</li> <li>• Dynamic memory allocation operators (new and delete)</li> <li>• Introduction to reference variables</li> <li>• Simple C++ program (using cin and cout)</li> <li>• Classes and Objects</li> <li>• Access specifiers</li> <li>• Defining Data members and Member functions, passing object to function, function returning object</li> <li>• Array of objects</li> <li>• Usage of 'this' pointer [ Ref. book 1- Chapter 2] [ Ref. book 2- Chapter 2,3,5]</li> </ul>	
<b>Unit 2: Functions</b>	<b>18</b>
<p><b>3. Function in C++</b></p> <ul style="list-style-type: none"> <li>• Introduction</li> <li>• Call by reference, return by reference</li> <li>• Function with default arguments, function overloading</li> <li>• Inline function (defining inline function, making class function inline)</li> </ul>	



<ul style="list-style-type: none"> <li>• Static class members and static member function</li> <li>• Friend functions and friend class [ Ref. book 1- Chapter 3] [ Ref. book 2- Chapter 4]</li> </ul> <p><b>4. Constructors and destructors</b></p> <ul style="list-style-type: none"> <li>• Constructors             <ul style="list-style-type: none"> <li>➤ Definition</li> <li>➤ Rules of defining constructor</li> <li>➤ Invoking constructor</li> <li>➤ Types of constructor</li> <li>➤ Multiple constructors in a class</li> </ul> </li> <li>• Destructor             <ul style="list-style-type: none"> <li>➤ Definition</li> <li>➤ Rules for writing destructor function</li> </ul> </li> </ul> <p>[ Ref. book 2- Chapter 6]</p> <p><b>5. Operator overloading</b></p> <ul style="list-style-type: none"> <li>• Concepts and rules</li> <li>• Overloading Unary and Binary operators</li> <li>• Overloading using member and friend function</li> <li>• Overloading insertion and extraction operator(&lt;&lt;,&gt;&gt; operators) [ Ref. book 1- Chapter 8] [ Ref. book 2- Chapter 7]</li> </ul>	
---	--

<p><b>Unit 3: Inheritance</b></p>	<p>8</p>
<p><b>6. Inheritance</b></p> <ul style="list-style-type: none"> <li>• Inheritance             <ul style="list-style-type: none"> <li>➤ Definition</li> <li>➤ Creating a derived class</li> <li>➤ Types of inheritance with examples (use of protected keyword)</li> </ul> </li> <li>• Constructors and destructor in derived classes</li> <li>• Pointer to derived class</li> <li>• Virtual base classes, Virtual functions and Pure virtual function</li> <li>• Abstract base class (definition with example) [ Ref. book 1- Chapter 9] [ Ref. book 2- Chapter 8,9]</li> </ul>	



<b>Unit 4: Input/output Handling</b>	<b>8</b>
<p><b>7. Managing Input and Output using C++</b></p> <ul style="list-style-type: none"> <li>• C++ stream classes</li> <li>• Formatted and unformatted console I/O ( get(),getline(),read(),put(),write() and ios formatting functions )</li> <li>• Manipulators and user defined manipulators [ Ref. book 1- Chapter 12] [ Ref. book 2- Chapter 10]</li> </ul> <p><b>8. Working with files</b></p> <ul style="list-style-type: none"> <li>• File stream classes and methods</li> <li>• File operations on text files and binary files</li> <li>• Random access files [ Ref. book 1- Chapter 12] [ Ref. book 2- Chapter 11]</li> </ul>	

<b>Unit 5: Templates and Exception Handling</b>	<b>6</b>
<p><b>9. Templates</b></p> <ul style="list-style-type: none"> <li>• Introduction to templates</li> <li>• Class templates, function templates and overloading of function templates</li> <li>• Templates with multiple parameters [ Ref. book 1- Chapter 14] [ Ref. book 2- Chapter 12]</li> </ul> <p><b>10. Exception Handling in C++</b></p> <ul style="list-style-type: none"> <li>• Introduction</li> <li>• Exception handling mechanism (try, catch and throw primitives)</li> <li>• Multiple catch statements</li> <li>• Nested try catch block</li> <li>• Rethrowing Exceptions [ Ref. book 1- Chapter 14] [ Ref. book 2- Chapter 13]</li> </ul>	

\*Contact hours – 12 hours

**Recommended Books:**

1. Robert Lafore ,*Object Oriented Programming with C++*, 4<sup>th</sup> Edition pearson
2. E. Balagurusamy,*Object Oriented Programming with C++*,5th Edition Mc Graw Hill
3. Herbert Schildt ,*The Complete Reference C++* , 4<sup>th</sup> Edition TMH Pub.

