COURSE CODE	COURSE TITLE	L	T	P	C
1151IT303	OBJECT ORIENTED PROGRAMMING LAB	0	0	2	1

Course Category: Program Core

a. Preamble:

In this course, programs will be implemented based on lab syllabus that is related to content which is given in theory and executed in C.

A. Prerequisite Courses:

Sl. No	Course Code	Course Name
1	1150CS201	Problem Solving using C

B. Related Courses:

Sl. No	Course Code	Course Name
1	1156IT601	Minor Project
2	1156IT701	Major Project

e. Course Outcomes:

Students undergoing this course are able To

CO Nos.	Course Outcomes	Knowledge Level (Based on revised Bloom's Taxonomy)
1	Apply the different data structures for implementing solutions to practical problems	S3
2	Develop recursive programs	S3
3	Develop Programs for Searching and Sorting	S3

f. Correlation with Programme Outcomes :

H- Strong; M-Medium; L-Low

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1.	H		M		H							
CO2.	H		M		H							
CO3	H	H	M	H								

g. Course Content

LIST OF EXPERIMENTS:

CYCLE I

S. No	Experiment name
1	Design C++ classes with static members.
2	Design C++ classes with default arguments, friend functions.
3	Implement complex number class with necessary operator overloading
4	Implement Matrix class with dynamic memory allocation and necessary methods. Give proper constructor, destructor, and copy constructor, and overloading of assignment operator.
5	Overload the new and delete operators to provide custom dynamic allocation of memory.

CYCLE II

6	Develop templates of standard sorting algorithms such as bubble sort, insertion sort.
7	Design stack and queue classes with necessary exception handling.
8	Design C++ Classes with Concepts of Inheritance.
9	Design C++ Program as virtual Functions and virtual Base Class.
10	Design C++ Classes with necessary File Handling Concepts.(Sequential and random merge sort, and quick sort. Access)

LIST OF EQUIPMENT FOR A BATCH OF 30 STUDENTS:

Stand alone desktops with C/C++ compiler 30 Nos. (or)

Server with C/C++ compiler supporting 30 terminals or more.

h. Learning Resources

i. Text Book

1. Fundamentals of Programming C++ (Richard L. Halterman).
2. Programming Abstractions in C++ (Eric S. Roberts)

ii. Reference Books

- **How to Design Classes (Matthias Felleisen, et al)**.
- *Programming -- Principles and Practice Using C++*.

iii.Online Resources

- http://www.byte-notes.com/oop-concepts-c
- https://www.hscripts.com/tutorials/cpp/cpp-oops-concepts.php

http://en.cppreference.com/w/cpp/concept