Reg. No.										

Manipal Institute of Technology, Manipal

(A Constituent Institute of Manipal University)

III SEMESTER B.TECH (COMPUTER SCIENCE AND ENGINEERING)

END SEMESTER EXAMINATIONS, NOV/DEC 2015

SUBJECT: OBJECT ORIENTED PROGRAMMING USING C++ [CSE 207]

(OLD SYLLABUS)

REVISED CREDIT SYSTEM
Date: 05-12-2015

Time: 3 Hours MAX. MARKS: 50

Instructions to Candidates:

- Answer **ANY FIVE** questions.
- Missing data, if any, may be suitably assumed.

1A.What is a reference variable? Give the syntax of declaring a reference variable. Explain with an example, how do you achieve call by reference using reference variable? 4M
1B. What is *this* pointer? Where and why does the compiler insert it implicitly? Explain. 3M
1C. How do you define an inline function? Explain with syntax and example. What are the situations in which a compiler ignores the request for inline function? 3M

2A. Write a complete C++ program that counts the number of times the character 'e' appears in the text file *letter.txt*, and prints that number to count. You should write all the necessary code in main; do not define any of your own functions. Make sure to include the necessary header files, and check that *letter.txt* has been opened successfully. 4M
2B. Declare a class called as Student which has name and admission number as data members. Include member functions, setData and showData to read and display the values of data members. With the help of this class show how to write the student record into binary file and display it. 4M
2C. What is a function prototype? Why is prototyping important? 2M

3A. Briefly explain the 3 types of constructors with an example for each. 3M
3B. What are the different ambiguities that arise in the case of multiple inheritance? Mention the different methods to resolve this. 3M
3C. How do you implement the following types of Inheritance? Explain with an example for each.
(i) Multilevel Inheritance
(ii) Hierarchical Inheritance 4M

4A. Explain the mechanism of virtual functions with suitable example. 4M
4B. Create an Invoice class that contains an item name and its price. Include member functions for setting the data members and displaying. Create an ItemException class, that holds an item name and an error message. While setting the data member values of the Invoice class, if the price is below Rs.10.00, then create an ItemException object with the

item name and “price too low” as the error message and throw it. This should be caught by the catch handler, which should display the details of the ItemException object. Write a main() function that declares an array of Invoice objects. The size of the array is entered by the user run time. If the ItemException object is thrown, make the user to enter new data for the invalid Invoice object of the array. 4M

4C. In what order, the constructors and destructors are invoked in the case of derived classes? Illustrate with suitable example. 2M

5A. What is an abstract base class? What are the uses of abstract base class? Illustrate with the help of a C++ program an abstract class and concrete class. 4M

5B. What is a friend function? Illustrate with an example, how do you make a global non-member function a friend of class. 4M

5C. What are the differences between a text file and a binary file? 2M

6A. Define a class Date with three private data members: day, month and year. Write default and parameterised constructors. Overload the operators <<, >> to read and print Date object. Overload == to compare two dates and return true/false (as a member function). Overload + operator to add an integer representing number of years to a Date object and return a Date object. (Eg. 5 + d1, where d1 is object of class Date). Write a destructor that sets values to zero for all three variables. Test this class in main function. 5M

6B. What is a namespace? How does namespaces help in preventing global namespace pollution? Explain with syntax and suitable example. 3M

6C. How do you define a new user defined manipulator? Explain with the help of an example. 2M