U.S.N ⬚⬚⬚⬚⬚⬚⬚⬚⬚⬚

P.E.S. College of Engineering, Mandya - 571 401
(An Autonomous Institution affiliated to VTU, Belgaum)
Fourth Semester, B.E. - Computer Science and Engineering
Semester End Examination; June - 2016
Object Oriented Programming with C++

Time: 3 hrs   *Max. Marks: 100*

Note: Answer **FIVE** full questions, selecting **ONE** full question from each unit.

UNIT - I

1 a. State and explain the basic concepts of OOPs with example. 5

 b. How name space helps in preventing the pollution of global namespace? 5

 c. Write a program to create a class called STUDENT with data members as st_name, st_Regno, test1, test2 and test3 marks. Create an array of '*n*' objects of this class and display all the information of the students along with the final test marks. (Final test marks is the average of best 2 Test marks). 10

2 a. Define function overloading. Write a program to define three overloaded functions to find the largest among 3 values. 7

 b. Define friend member function and friend non-member function. Explain each with an example. 7

 c. Define inline function. Write an inline function to swap two numbers. 6

UNIT - II

3 a. Define constructors and destructors. Explain the different types of constructors with example. 9

 b. Define dynamic memory allocation. Write a C++ program to read and print the array of floating point numbers using new and delete operators for allocating and de-allocating memory for array. 6

 c. Explain set-new-handler( ) function with example. 5

4 a. Define inheritance. Explain different types of inheritance. 6

 b. Write a C++ program to create a base class called student, from this inherit a new class called exams. Containing marks 1, marks 2 and marks as data members. Create another base class called sports having sports_grade as its data member. Create a leaf class called award which is derived from exams and sports classes. Use suitable member functions in all these classes. 8

 c. Define function overriding. Illustrate function overriding with example. 6

UNIT - III

5 a. Justify the need of virtual functions. Illustrate the same with example. 10

  b. Define pure virtual functions. Create a class called GEOFIG with a floating point data member. Include member function to access and display that data. Also include a pure virtual function compute_area( ) to calculate area. Derive two classes CIRCLE and SQUARE. Write a program to make objects of CIRCLE and SQUARE. Display their areas. 10

6 a. Write and explain the syntax of the functions that are used in the manipulation of file pointers. 6

  b. With example explain predefined manipulator functions. 7

  c. Explain various flags and functions associated with error handling. 7

UNIT - IV

7 a. Define operator overloading list the rules of operators overloading. 8

  b. Write a C++ program to create a class called complex with the data member's real_part and img_part. Define the following member functions :

    i) To input data

    ii) Overload '+' operator to add two complex numbers

    iii) Overload '-' operator to subtract a complex number from another complex number.

    iv) To output the contents of the object. 12

8 a. Write a C++ program to illustrate overloading of increment and decrement operator (both post fix and prefix). 10

  b. With example explain how to overload '[ ]' and pointer – to – member '$\rightarrow$' operator with example. 10

UNIT - V

9 a. Write a C++ program to implement stack for integers and float point numbers using templates. 10

  b. Define function template and class templates. Write a note on STL. 10

10 a. Define exception handling. With an example explain how exceptions are handled in C++? 10

  b. Write a C++ program to implement bubble sort for integers and floating point numbers using templates. 10

∗ ∗ ∗ ∗