Code: DE122/DC106   Subject: Object Oriented Programming with C++

DipIETE – ET/CS *(NEW SCHEME)*

Time: 3 Hours   **DECEMBER 2014**   **Max. Marks: 100**

PLEASE WRITE YOUR ROLL NO. AT THE SPACE PROVIDED ON EACH PAGE IMMEDIATELY AFTER RECEIVING THE QUESTION PAPER.

NOTE: There are 9 Questions in all.
- **Question 1 is compulsory and carries 20 marks. Answer to Q.1 must be written in the space provided for it in the answer book supplied and nowhere else.**
- **The answer sheet for the Q.1 will be collected by the invigilator after 45 minutes of the commencement of the examination.**
- **Out of the remaining EIGHT Questions answer any FIVE Questions. Each question carries 16 marks.**
- **Any required data not explicitly given, may be suitably assumed and stated.**

Q.1   **Choose the correct or the best alternative in the following:**   **(2×10)**

   a.   Which of the following operator can be overloaded?

   (A) size of operator   **(B)** Dot operator ('.')
   (C) scope resolution operator ('::')   **(D)** Multiplication operator ('∗')

   b.   The operator >> in C++ is called _____.

   (A) an extraction operator   **(B)** a put to operator
   (C) a catch operator   **(D)** None of these

   c.   How constructor differs from destructor

   (A) constructors can be overloaded but destructors can't be overloaded
   (B) constructors can take arguments but destructor can't
   (C) there is no difference
   (D) both **(A)** and **(B)**

   d.   How many 'catch' one can associated with a **try**?

   (A) Only one   **(B)** More than one
   (C) Two   **(D)** None of these

   e.   An exception is caused by

   (A) a hardware problem   **(B)** a problem in the operating system
   (C) a syntax error   **(D)** a run time error

   f.   The for construct **for(;;)** will lead to

   (A) infinite loop
   (B) syntax error
   (C) unexplained termination of code execution
   (D) no evident change in code execution

g. How many destructors can a class have?

   (A) any number   **(B)** 1
   (C) 2   **(D)** None of these

h. The function overloading and operator overloading resolve the necessary conversions at

   (A) Compile time   **(B)** Execution time
   (C) Linking time   **(D)** All of these

i. A value that is automatically passed to a function when no explicit argument is specified in the function call is called

   (A) Call-by value   **(B)** Default argument
   (C) Constant argument   **(D)** Virtual function

j. Inheritance is a way to _____.

   (A) organize data.
   (B) pass arguments to objects of classes
   (C) add features to existing classes without rewriting them
   (D) improve data-hiding and encapsulation

Answer any FIVE Questions out of EIGHT Questions.
Each question carries 16 marks.

Q.2 a. Distinguish between Procedure-oriented programming and Object-Oriented Programming. **(8)**

   b. Write a program in C++ which calculates the factorial of a given number. **(8)**

Q.3 a. What is the main advantage of passing arguments by reference? Explain this with an example. **(4)**

   b. What does 'this' pointer point to? Explain. **(4)**

   c. Write a program in C++ to sort an array of positive integers. **(8)**

Q.4 a. How do the properties of following two derived classes differ? **(4)**
      (i) class X : public A{//..}
      (ii) class Y : private A{//..}

   b. Explain the characteristics of static class members. **(4)**

   c. Is it possible that a function is friend of two different classes? If yes, then how it is implemented in C++? **(8)**

Q.5 a. Why do we need constructors? **(4)**

b. Differentiate between Default constructor and copy constructor using suitable example. **(4)**

c. Explain the concept of operator overloading. Write a program to overload the operator '+' for complex numbers. **(8)**

Q.6 a. What does inheritance mean in C++? What are the different forms of Inheritance? **(8)**

b. What is multiple inheritance? Write a program that explains how to pass parameters to the constructors of base classes in multiple inheritance. **(8)**

Q.7 a. Differentiate between late and early binding. **(4)**

b. How is polymorphism achieved at run time? Explain with appropriate example. **(8)**

c. How is exception handling implemented in C++? **(4)**

Q.8 a. Differentiate between function overloading and function templates. Explain using examples for both. **(8)**

b. Write a function template to find a maximum value from an array. **(8)**

Q.9 a. Explain the I/O stream hierarchy in C++. **(8)**

b. Explain the following functions(with example) for manipulating file pointers:
seekg(), seekp(), tellg(), tellp() **(8)**