

GUJARAT TECHNOLOGICAL UNIVERSITY
M.C.A.- SEMESTER – II • EXAMINATION – WINTER 2012

Subject code: 2620002**Date: 26-12-2012****Subject Name: Object-Oriented Programming Concepts
and Programming (OOCp)****Time: 02:30 pm – 05:00 pm****Total Marks: 70****Instructions:**

1. Attempt all questions.
2. Make suitable assumptions wherever necessary.
3. Figures to the right indicate full marks.

- Q.1 (a) Do as directed: 07**
1) Which storage class data member is having single copy of variables for all objects? Describe with appropriate example.
2) Explain with example : Pointer to Constant & Constant Pointer
- (b) Answer in short 07**
1) What are the different access specifiers available in C++?
2) Explain the term Containership. Also discuss how it is different than inheritance.
- Q.2 (a) What is a friend function? Explain different usages of friend function and give example where friend function is used as a bridge. 07**
- (b) What are the different ways to overload an operator for a user-defined class? When is it must to overload an operator as a friend function? Support your argument with an example. 07**
- OR**
- (b) What is run-time polymorphism? Explain how it is achieved in C++. 07**
- Q.3 (a) What are templates? What are function and class templates? What is template specialization? Explain the concept of template argument deduction. 07**
- (b) What do you mean by namespaces and why are they required? Discuss the name conflict problem. Explain the following ways of using names defined inside the namespace. 07**
 - fully qualified names
 - using 'using' declaration
 - using 'using' directive
- OR**
- Q.3 (a) How is Exception Handling mechanism useful? Explain how it is implemented in C++. Explain usage of terminate() and unexpected(). 07**
- (b) Explain very briefly the following terms: 07**
1) destructor
2) export
3) explicit
4) reinterpret_cast
5) this
6) typename
7) virtual base class

Q.4 (a) Create two classes Rupee and Dollar to store the value of money. Rupee stores money in Rs and Paisa while Dollar stores money in Dlr and Cents. Write a C++ program that reads data into objects of class Dollar and class Rupee. Add Rupee and Dollar using + and display the resulting amount in terms of Rupee or Dollar using << operator. **07**

(b) What are manipulators? How are they different than ios formatting functions? How can one create his/her own manipulator? **07**

OR

Q.4 (a) Define a class Computer. Also define classes of Main, Mini and Micro inheriting from that. Main is further inherited into Super and Non-Super, while Micro is inherited into PC and HandHelds. Define an array which contains 10 different types of Computers. Define a function ReadDetails() in all above classes. Write a C++ program to create this single array, read details of different computers and display them. Details should be displayed based on the type of the computer. **07**

(b) What is file opening mode? Describe the various file opening modes available in C++. Explain how a file pointer can be moved / managed in a random file for read / write operations. **07**

Q.5 (a) What is RTTI? Explain how typeid and dynamic_cast are used for RTTI. **07**

(b) What is STL? What is Generic Programming? Explain the main components of C++ STL. **07**

OR

Q.5 (a) I. State true/false **05**

1) Inheritance means the ability to reuse the data values of one object by other objects.

2) cin and cout are built-in functions.

3) The overloaded operator must have at least one operand that is user-defined type.

4) If any derived class contains a constructor with one or more arguments, then it is mandatory for the base class to have a constructor and pass the arguments to the derived class constructor.

5) "this" is an implicit argument to all member functions.

II. Give Error/Output

02

```
#include <iostream.h>
#include <conio.h>
class base
{
    public :
        virtual void fun1() { cout << "\n In base::fun1"; }
        void fun2() { cout << "\n In base::fun2"; }
};
class derived : public base
{
    public :
```

```

        void fun1() { cout << "\n In derived::fun1";      }
        void fun2() { cout << "\n In derived::fun2";      }
};

void main()
{
    base *ptr1,*ptr2;  base b;  derived d;
    ptr1 = &b;    ptr2 = &d;
    ptr1->fun1();
    ptr1->fun2();
    ptr2->fun1();
    ptr2->fun2();
}

```

- (b)** What is polymorphism? Explain in detail different ways of achieving compile-time polymorphism in C++. **07**
