U.S.N.										

B.M.S. College of Engineering, Bengaluru-560019

Autonomous Institute Affiliated to VTU

December 2023 Supplementary Examinations

Programme: B.E. **Semester: III**
Branch: Information Science and Engineering **Duration: 3 hrs.**
Course Code: 22IS3PCOOP **Max Marks: 100**
Course: Object Oriented Programming using C++

Instructions: 1. Answer any FIVE full questions, choosing one full question from each unit.
2. Missing data, if any, may be suitably assumed.

UNIT - I

1 a) Differentiate Object Oriented and Procedure Oriented Programming. **05**

b) Multiple names can be given to an existing variable. Justify your answer with proper reasoning and examples. **05**

c) Write a program to create a class room that has three member variables length, breadth and height. Member functions to read, display compute area and compute volume of the room. Member functions should be defined outside the class. Write main function to create 15 room objects and calculate area and volume of those objects. **10**

UNIT - II

2 a) Is it possible to reduce the overhead of the function call? Justify your answer with an example. **05**

b) **05**

```
Class time1
{
  int hr,mi,sec;
….
}
int main()
{
 time1 t1,t2,t3
….
   T3.add(t1,t2);
  add(t1,t2,t3);
....
}
```

Complete the given code with appropriate functions to read, display and add two time objects that supports the function call given in main.

c) Complete the following code to support following ways of creating the objects of a class Demo with two member variables x, y. **05**

```
int main()
{
   Demo d1;
   d1.display();
   Demo d2(10,5);
   d2.display();
   Demo d3(50);
   d3.display();
   Demo d4(d2);
   d4.display();
}
```

d) Write a program to create a class demo which has 2D array as member variable. Allocate memory and initialize the 2D array using dynamic constructors. Write destructors to free the memory. **05**

UNIT - III

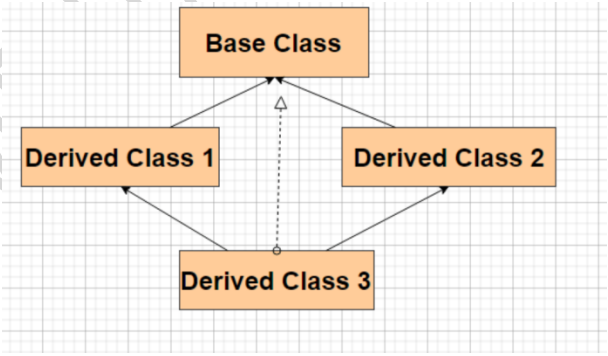
3 a) What are the rules for overloading operators? What are the operators that cannot be over loaded? **05**

b) Write a program to input a complex number and display the same by overloading << and >> operator. **07**

c) Write a program to overload pre-increment and post-increment for a class example with two member variables. **08**

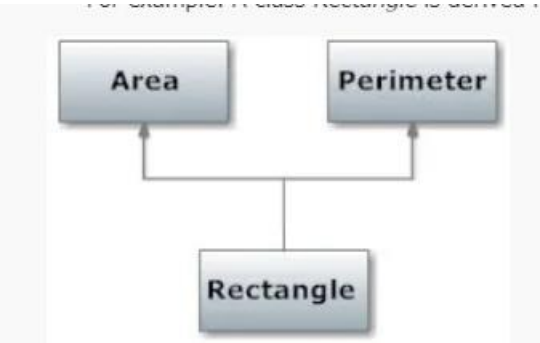
OR

4 a) **05**

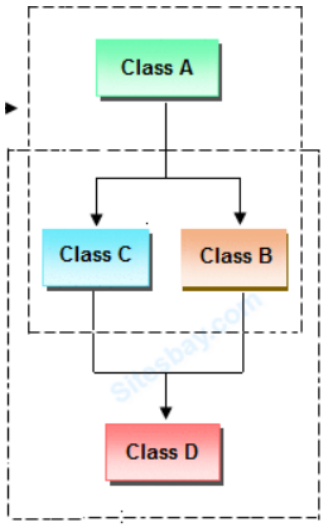


Analyze the error which will occur in the above relationship. Provide a solution for the same with a C++ program.

b) Identify the type of Inheritance given in the diagram and write a program to demonstrate the same. **05**



c) Develop a program to implement the inheritance given in the diagram. **10**



UNIT - IV

5 a) Write a pure virtual function, output() for class A which acts as Base class for C and S. Print the different messages in each class. Demonstrate it with a main function. **05**

b) Write a program with a Class Complex that has two member variables to demonstrate pointer to objects. **05**

c) Illustrate with a program, pointers of base class to access derived class methods. **05**

d) i) Can we declare a static function as virtual. Justify the answer. **05**
ii) Can the member functions in base and derived class have the same name. Illustrate with an example program.

OR

6 a) Develop a program to copy content of one file to another file. **10**

b) Predict the outputs of the following code: **05**
i) inf.seekg(14, std::ios::cur);
ii) inf.seekg(-18, std::ios::cur);
iii) inf.seekg(22, std::ios::beg);
iv) inf.seekg(24);
v) inf.seekg(-28, std::ios::end);

c) Write C++ code for displaying the following output **05**

a) 0.1, 1, 1.23457e+009
b) 0.100000, 1.000000, 1234567936.000000
c) 1.000000e-001, 1.000000e+000, 1.234568e+009
d) 0.100, 1.000, 1234567936.000
e) 0.1000000014901161
f) ******3445
g) ******34******45

UNIT - V

7	a)	Assume an application that has a Class Emp with id, name and salary as member variable. Write read method that accepts values, throws user defined exceptions when salary entered is less than 0.	**10**
	b)	Develop function template for finding the minimum of elements contained in an array. Demonstrate for two different data types.	**05**
	c)	Develop a class template that demonstrates two generic data types being passed for the Class Test.	**05**

******