2022

(July)

MCA

Paper Code: MCA-0801

(Object Oriented Programming and C++)

Full Marks : 75

Time : 3 Hours

The figures in the margin indicate full marks for the questions

Attempt **one** question from each **UNIT**

UNIT – I

1. a) What is the significance of destructor? Describe *any two* features of Object Oriented Programming. (2+4=6)

    b) Differentiate between *call by reference* and *call by address*. What is *inline* function? How is it useful? (4+2+2=8)

    c) Briefly discuss the implication of passing default arguments to constructors with an example. What are the advantages of using *new* and *delete* operators as compared to the *malloc()* and *calloc()*? (3+3=6)

2. a) When does it become mandatory to have a copy constructor? What are the various scenarios in which it is called? Differentiate between C and C++. (2+2+4=8)

    b) How is friend function different from a member function? Can non-static member function access static data members? Justify your answer. (3+3=6)

    c) Explain the concept of arrays of objects with a suitable example. Comment on the code below: (4+2=6)

    P *t=new P[5];
    delete t;

UNIT-II

3. a) What is Standard Template Library (STL)? What are its basic components? With suitable example, explain what happens if an exception is thrown but not caught. (1+3+5=9)

    b) What is the significance of *auto_ptr*? Illustrate rethrowing an exception with an illustrative example. (2+3=5)

c) How can you restrict a function to throw certain exceptions? What are the differences between overloaded function and function template? (2+4=6)

4. a) Describe the use of non-type parameters for templates with an example. How can you provide default arguments for template parameters in class templates? (4+4=8)
   b) What is the difference between error and exception? Briefly explain, how exceptions are handled in C++. (1+4=5)
   c) Write a code to demonstrate a class template with static data member. With suitable example, briefly discuss standard exceptions in C++. (3+4=7)

UNIT-III

5. a) What is the significance of *this* pointer? Differentiate between *overloading* unary and binary operators. List out the operators which cannot be overloaded. (2+4+2=8)
   b) What are impacts of *vtable* and *vptr* in binding? Illustrate data conversion from one class type to another class type. (2+4=6)
   c) Discuss nested class with an illustrative example. What do you understand by object slicing? (4+2=6)

6. a) Explain protected inheritance. How can you call the parameterized constructor of base class in derived class constructor? (3+3=6)
   b) What do you understand by upcasting? What is the ambiguity in hybrid inheritance? How can it be resolved? (2+3+3=8)
   c) Briefly discuss on: (3+3=6)
      i. Abstract class
      ii. Virtual destructor

UNIT-IV

7. a) What do you understand by UML? What are the advantages of using UML? Explain the different elements of a Use Case. (1+2+4=7)
   b) What are the uses of *getline()* and *write()* functions? (2+2=4)
   c) Discuss error handling in file operations. (4)

8. a) What do you understand by Object Oriented Methodology (OOM)? Briefly explain the stages of OOM. (1+5=6)
   b) What is the difference between *seekp()* and *seekg()*? Briefly describe the various components in sequence diagrams. (1+4=5)
   c) Write a C++ code to open a text file and replace all the vowels in lowercase to uppercase and vice-versa leaving other character as it is. (4)

◆◆◆◆◆◆◆◆◆◆