

Roll No.

Total No. of Questions : 12]

[Total No. of Pages : 03

Paper ID [C1419]

(Please fill this Paper ID in OMR Sheet)

B.Sc. BI (402) (Sem - 4th)

OBJECT ORIENTED PROGRAMMING USING C ++

Time : 03 Hours

Maximum Marks : 75

Instruction to Candidates:

- 1) Section - A is **Compulsory**
- 2) Attempt any **Nine** questions from Section B.

Section - A.

(15 × 2 = 30)

Q1)

- a) List at least four new operators by C ++ which aid OOP.
- b) Why is an array called derived array type?
- c) Write a function using reference variables as arguments to swap the values of a pair of integers.
- d) What are the advantages of function prototypes in C ++?
- e) How do structures differ in C and C ++?
- f) What is a friend function? What are the merits and demerits of using friend function?
- g) What is a constructor? Is it mandatory to use constructors in a class?
- h) What is operator overloading?
- i) What are the different forms of inheritance? Give an example for each.
- j) Distinguish between overload functions and function templates.
- k) What is an exception?
- l) What is a containership? How does it differ from inheritance?

J-8210[S-9700365]

P.T.O.

- m) What do you mean by dynamic binding? How is it useful in OOP?
- n) What is object oriented programming? How is different from the procedure oriented programming?
- o) List a few areas of application OOP methodology.

Section - B

(9 × 5 = 45)

- Q2)** a) Define the following terms with examples:
- (i) Abstract data type
 - (ii) Polymorphism
 - (iii) Inheritance
 - (iv) Templates
- Q3)** Design an EMPLOYEE base class. Define all its possible methods and data structures. Through inheritance mechanism, create one class namely MANAGER. Implement its data structure and important methods. Assume that you are making this design for the purpose of making a salary statement. Observe the following while designing classes:
- Clearly indicate public and private classes
 - Design constructors in each class and explain its purpose
 - Identify data structure and methods which can be inherited
 - Implementation should be in C++
- Q4)** Prepare object diagrams showing at least 10 relationships among the following object classes. Include associations, aggregations and generalizations. Use qualified associations and show multiplicity balls in your diagrams. You do not need to show attributes or operations. Use association names where needed. As you prepare the diagrams you may add additional object classes:
School, playground, principal, classroom, book, student, teacher, cafeteria, rest-room, computer, desk, chair, door.
- Q5)** Prepare a data flow diagram for computing the volume and surface area of a cylinder. Inputs are height and radius of cylinder. Outputs are volume and surface area. You can use any arbitrary formula for demonstration. Discuss several ways of implementing the DFD.

- Q6)** Design a template class for implementation of BINARY TREE.
- Q7)** Implement the following class using C++ and any other non-object-oriented language: Sorted_dictionary
- Q8)** Write short notes on the following:
- (a) Reusability
 - (b) Extensibility
 - (c) Robustness
- Q9)** Rephrase the following requirements to make them more precise. Remove any design decisions posing as requirements.
- A system for distributed electronic mail over a network is needed. Each user of the system should be able to send mail /Tom any computer account and receive mail on one designated account. There should be provisions for answering or forwarding mail, as well as saving messages in files or printing them. Also users should be able to send messages to several other users at once through distribution lists. Each computer on the net should hold any messages destined for computers which are down.
- Q10)** Write code to implement the following:
- One-to-many association, which is traversed in the direction /Tom one to many. The association considered unordered Code should be in C++.
- Q11)** Design a template class for implementation of a LINKED LIST.
- Q12)** Implement the following class using C++ and any other non-object oriented language polygon.

