This question paper contains **2** printed pages]

GF—04—2023

FACULTY OF SCIENCE AND TECHNOLOGY

B.Sc. (CS) (Second Year) (Third Semester) EXAMINATION

APRIL/MAY, 2023

(CBCS/Revised Pattern)

COMPUTER SCIENCE

Paper BCS-301

(Object Oriented Programming)

(Wednesday, 19-4-2023) **Time : 2.00 p.m. to 5.00 p.m.**

Time—Three Hours *Maximum Marks*—75

N.B. :— (*i*) *All* questions are compulsory.

(*ii*) Figures to the right indicate full marks.

1. Attempt any *five* of the following : 15

(*a*) Explain applications of OOPs.

(*b*) Explain the concept of manipulators.

(*c*) Explain concept of static member function.

(*d*) Explain the use of constructor.

(*e*) Explain various data types used in C++.

(*f*) Discuss structure of C++ program.

(*g*) Write a program in C++ for addition and subtraction of two numbers.

2. Attempt any *three* of the following : 15

(*a*) Explain the concept of copy constructor in C++.

(*b*) Write a program in C++ to describe concept of function overloading.

(*c*) Discuss concept of call by reference in detail.

(*d*) Write a C++ program to describe concept of class and object.

(*e*) Explain concept of friend function in detail.

P.T.O.

3. Attempt any *three* of the following : 15

(*a*) Explain concept of default argument with example.

(*b*) Explain if and if-else statement used in C++ with syntax and example.

(*c*) Write a C++ program to describe Multiple inheritance.

(*d*) Explain various stream classes used in C++.

(*e*) Write a program in C++ to reverse the digits of a given number.

4. Attempt any *three* of the following : 15

(*a*) Explain the concept of class and object in detail.

(*b*) Explain the concept of pointer to object with example.

(*c*) Write a C++ program to describe concept of Binary operator overloading.

(*d*) Explain concept of STL in detail.

(*e*) Write a C++ program to describe concept of for loop.

5. Write short notes of any *three* of the following : 15

(*a*) Virtual base class

(*b*) Friend function

(*c*) Static member function

(*d*) Command line arguments

(*e*) Operator overloading.