

## Object Oriented Programming With C++ (2140705)

### List of Practicals

1. Write a program to illustrate uses of enumerations in C++.
2. Write a program to demonstrate call by reference and return by reference.
3. Write a program to illustrate inline functions.
4. Write a program to illustrate function overloading.
5. Write a program to illustrate use of default arguments in C++.
6. Create a class coordinate containing x, y and z private variables. Perform operations for incrementing, adding and comparing object(s) by overloading +, += and == operators respectively. Define necessary functions to set and display the variables.
7. Write a program to find out sum of two private data members x and y of two classes A and B using a common friend function. Assume that the prototype for both the classes will be ***void sum (A, B);***
8. Create two classes X and Y containing private variables x and y respectively. Using a common friend function, perform multiplication operation between x and y.
9. Write a program to illustrate use of virtual functions.
10. Declare a class called logic\_gate to represent logic gates. The class has three data members - input1, input2 and input3 to represent three inputs to the logic gate. The class also has a virtual function member called get\_gate\_output. Derive two classes from the base class logic\_gate, namely, and\_gate and or\_gate to represent 'logical and gate' and 'logical or gate' respectively. Define function get\_gate\_output in both of these classes to get the output of the gate. Show use of above classes and functions to demonstrate dynamic polymorphism in function main.
11. Write a program to demonstrate copy constructor, default constructor and parameterized constructor.
12. Declare a class called bird having private data members name and weight. Define following functions: - default constructor for reading data members from key board - overloaded constructor with two arguments to be used for initialization of data members. - display function to display data members. - overloaded member operator >= to compare weight of two bird objects, returning false if weight of first bird object is less than that of the second & true otherwise. Define main to illustrate use of above functions.
13. Write a program to demonstrate use of destructors in C++.
14. Declare a class called book\_details to represent details for a book, having data members like title, author, edition, price and no\_of\_copies\_available. Define following functions:
  - constructor(s)
  - display to display all data members
  - find\_books to find and display details of all books having price less than Rs. 250
  - main to create an array of book\_details and to show usage of above functions.
15. Define a class complex with real and imaginary as two data member, add necessary constructors and member function to initialize and display data of class. Class should overload the + operator to add two complex objects and return the results. Invoke the statements like C3=C1+C2 in main ().
16. Declare a class called book having members like book\_title, publisher and author\_name. Overload extractor and inserter operators ( >> and << ) for class book.
17. Write a program to overload binary + operator as a member function.
18. Write a program to illustrate type conversion from class type to basic type, one class type to other class type and basic type to class type.
19. Write a program to illustrate Scope Resolution operator (::).
20. Write a program to illustrate types of inheritance supported in C++.

21. Create two classes to illustrate use of method overriding in C++.
22. Write a program to illustrate virtual base class.
23. Illustrate the program to demonstrate runtime polymorphism.
24. Declare a class called `logic_gate` to represent logic gates. The class has three data members - `input1`, `input2` and `input3` to represent three inputs to the logic gate. The class also has a virtual function member called `get_gate_output`. Derive two classes from the base class `logic_gate`, namely, `and_gate` and `or_gate` to represent 'logical and gate' and 'logical or gate' respectively. Define function `get_gate_output` in both of these classes to get the output of the gate. Show use of above classes and functions to demonstrate dynamic polymorphism in function `main`.
25. Write a program to illustrate use of this pointer in C++.
26. Declare a class called `item` having data members `item_code`, `item_name`, `cost` and `discount`. Derive two classes from class `item`, namely `employee` and `customer`. The class `employee` has data members like `employee_code`, `employee_name` and `amount`. The class `customer` has data members like `customer_name` and `amount`. Define following functions for - initializing data members. - displaying the values of data members. - computing amount to be paid for a purchased item. Also define function `main` to create objects of both derived classes and to show usage of above functions.
27. Write a program that opens two text files for reading data. It creates a third file that contains the text of first file and then that of second file (text of second file to be appended after text of the first file, to produce the third file).
28. Write a program to copy the contents of a source file `student1.txt` to a destination file `student2.txt` character by character. Assume various file mode parameters in C++.
29. Write a program which uses command line argument to copy the contents of a file `A.txt` into another file `B.txt` by reversing case of the characters. E.g. File `A.txt`: `aBCd` File `B.txt`: `AbcD`
30. Write a program that reads a text file and creates another file that is identical except that every sequence of consecutive blank space is replace by a single space.
31. Declare a template class called `exam` having an array of generic type as a data member, named `elements[10]`. Define following generic (template) member functions:
  - `sort` to arrange elements in ascending order
  - `find_max` to find and return maximum from the arrayDefine `main` to illustrate usage of these functions to process two different types of data.
32. Write a function template for finding the minimum value contained in an array.
33. Write a program that illustrates the application of multiple catch statements.
34. Write a class template to represent a generic vector. Include member function to create the vector and to modify the value of a given element.
35. Write a program to handle exceptions in C++.
36. Write a program to create a class `distance` containing feet and inches. Using operator keyword, convert an object of class `distance` into total meters which is a float data type. (1 meter=3.28 feet)