

## Lab 3

1. WAP to create a class Teacher with data members teacher\_id, teacher\_name, department and subject\_taught and create member functions for reading and displaying data members. At least one of the member function should be defined outside the class.
2. Create a class called “time” with data member hour, minute and second. Initialize all the data member using constructor. WAP to add two time object using necessary member functions and display the result.
3. Create a C++ class called 'Person' with a constructor that initializes the name and age member variables. Implement a copy constructor that creates a new Person object by copying the name and age from an existing object. Show the usage of both constructors by creating an object named 'person1' with name 'Sunil' and age 25. Then, use the constructors to create another object named 'person2' with the same name and age as 'person1'. Print the details (name and age) of 'person2'.
4. Write a C++ program that defines a class called "Rectangle" with private member variables "length" and "width". The class should have the following member functions:
  - i. A default constructor that initializes both "length" and "width" to 0.
  - ii. A parameterized constructor that takes two arguments and initializes "length" and "width" accordingly.
  - iii. A function called "calculateArea" that calculates and returns the area of the rectangle.
  - iv. A function called "doubleSize" that takes a Rectangle object as an argument, doubles its length and width, and returns the modified object.In the main function, create a Rectangle object using the parameterized constructor with the values 4 and 5. Then, pass this object to the "doubleSize" function, and display the area of the modified rectangle.
5. WAP to show the destructor call such that it prints the message “memory is released”.
6. Write a program that manages an array of student objects. Each student object contains a name and age. The program should prompt the user to enter the number of students and their respective names and ages. Afterward, calculate and display the average age of all the students. Furthermore, find and print the

name of the oldest student in the array. Provide the code to implement this program, including the necessary class definition and the logic to calculate the average age and find the oldest student.