OOPD Laboratory (Syllabus)

Course instructor: Dr. Charulata Patil

Laboratory credits: 1

Teaching Scheme: Lab: 2 hrs/week

Examination Scheme: Continuous Assessment: 50 Marks, End Semester Exam: 50 Marks

Instructions:

- Feel free to enhance the problem statements and incorporate multiple OO or Java features. Justify your additions during evaluation.
- Note that the same assignment can be performed in multiple ways. Try to use maximum OO features.
- Implementation language: Java
- Evaluation: During regular lab hours or as per given deadline from time to time
- Submission: Instructions will be given later

Assignments:

1. Assignment 1: User input, Java data types, operators and control structures
   - ATM Machine Simulator: The program should allow a customer to perform simple banking operations like check balance, deposit, withdraw and transfer funds on the available balance depending on the type of customer. Allow or deny user operations depending on user type and operation using bitwise operators E.g. Check balance has value 1, transfer funds has value 8 etc.. User of type "Premium" has value 15 and he will be able to perform all operations.
   - Use: User input, data types, bitwise operations, assignment operations, control structures like if, switch, while etc.
   - Input: Balance, user type, operation, repeat? etc.
2. Assignment 2: Class and instance variables and methods
   - A library system maintains a record of books. Every book has title, price, edition etc. Maintain a record of at least 10 books.
   - Whenever a book is issued, the total count of issued books increases and it decreases after the book is returned.
   - Every student may issue or return a book.
   - At any point in time, print what percentage of total books are issued from the library.
3. Assignment 3: Constructors, constructor overloading and method overloading
   - Employees in an organization have id, name, salary, bonus, rating.
   - Every employee must have an id and name.
   - By default, the employee has salary Rs. 10000.00, bonus as 0 and rating as 1
   - Employee can have any other salary and/ or bonus amount.
   - Depending on the rating given to the employee, the bonus is calculated as bonus amount * rating
   - Create record of 10 employees in an array.
   - Ask the employer if bonus or rating or both for any employee is to be updated
   - Print the before and after salary details of every employee.
4. Assignment 4: Inheritance and method overriding
   - In a Vehicle Management System, every vehicle has brand, fuel, max speed, wheels and methods: displayInfo(), start(), stop().
   - Manual vehicle inherits from Vehicle, has fuel = None, Auto vehicle inherits from Vehicle, has mileage.
   - Any vehicle is either Manual vehicle or Auto vehicle.
   - Bike inherits from Auto vehicles has wheels=2 and has some other attributes and methods like displayInfo(), ride() etc.
   - Car inherits from Auto vehicles has wheels=4 and has some other attributes and methods like displayInfo(), honk() etc.
   - Demonstrate polymorphism.

5. Assignment 5: Generics
   - Implement a Generic Inventory Management System for a retail store.
   - The store deals with different types of items such as Electronics, Groceries, and Clothing.
   - Instead of creating separate inventory classes for each item type, the system should use Java Generics to create type-safe, reusable components
6. Assignment 6: Exception handling
   - Create an Employee leave management system.
   - Every Empolyee has id, name, total leaves, casual leaves, earned leaves etc.
   - Employee applies for leave. Allow this operation only if there is enough balance of that type of leave.
   - Manager is able to approve or reject leave application of the employee. The leaves should be updated accordingly.
   - Use assert, custom exceptions, throw, throws, try, catch, finally
7. Assignment 7: Multithreading
   - Create a multithreaded ticket booking system
   - There are limited number of tickets e.g. 100
   - A user can book only one ticket.
   - Many users e.g. 250 try to book tickets and few of them e.g. 15 cancel already booked ticket at the same time.
8. Mini project:
   - Choose any real-life problem statement and design and implement an OO system for the same
   - The project is to be completed in group of max size 4 students.
   - It must include
     - ➢ OO design diagrams
     - ➢ Data hiding
     - ➢ Inheritance and polymorphism
     - ➢ Packages and interfaces
     - ➢ Custom exceptions and exception handling
     - ➢ Other OO features