



COMSATS University Islamabad
Department of Computer Science
Midterm Examination FALL 2025

Class: BCT-3, BSE-2
Subject: CSC241- Object Oriented Programming
Instructor: Saif Ullah Ijaz

Marks: 25
Time: 90 Mins
Date: 28-October-2025

Instructions:

- All questions are self-explanatory and require no further explanation during exam time.
- Return the question paper along with the answer sheet.
- Attempt all questions.

[CLO1: Demonstrate fundamental principles and concepts of object-oriented programming.]

Question 1. Briefly explain the following:

[3*3=9 Marks]

- Explain with code how object references are passed to methods in Java
- Design a simple example of a copy constructor in Java and explain its importance.
- Describe what an immutable class is and how we can create an immutable class?

Question 2) For the following CryptoWallet class, mark each field and method with the appropriate access modifier. Also identify the static and final members if any. [3 Marks]

```
public class CryptoWallet {  
    // FIELD 1: Wallet owner's username - accessible to other classes in same package  
    /* Package Private */ String ownerName;  
  
    // FIELD 2: Blockchain wallet address - immutable after wallet creation  
    Private Final String walletAddress;  
  
    // FIELD 3: Wallet balance in crypto - only modifiable within this class  
    private double cryptoBalance;  
  
    // FIELD 4: Total wallets created on network - shared across all instances  
    Private Static int totalWallets;  
  
    // FIELD 5: Network name (e.g., "Ethereum") - same for all wallets, cannot change  
    Private static final String blockchainNetwork;  
  
    // FIELD 6: Transaction fee percentage - same for all, modifiable by admin  
    Private Static double transactionFee;  
  
    // METHOD 1: Constructor - should be callable from anywhere to create wallets  
    Public CryptoWallet(String owner, String address, double balance) {  
        // initialization code. Don't write method body.  
    }  
  
    // METHOD 2: Validate wallet address format - doesn't need object instance  
    Public static boolean isValidAddress(String address) {
```

```

    // validation logic. Don't write method body.
}

// METHOD 3: Update wallet balance - should only be called within this class
    Private void updateBalance(double amount) {
    // update logic. Don't write method body.
}

// METHOD 4: Get wallet info - accessible to subclasses (e.g., PremiumWallet)
    Protected String getWalletInfo() {
    return "Address: " + walletAddress + ", Owner: " + ownerName;
}
}

```

[CLO2: Design an Object-Oriented model for a real-world problem.]

Question 3. For the scenario given below do the following:

[5+8=13 Marks]

1. Provide the graphical representation of the system using class diagram
2. Provide the java implementation of the system that applies OOP principles properly, including inheritance, aggregation, and composition

Consider a Cricket Team Management System in which you need to manage a cricket team that consists of Players and a Coach, who are all team members. Every team member should have basic details such as their name, email, and base salary. For Players, the program should also record their role (Batsman, Bowler, or All-Rounder), and for the Coach, the program should record their specialization (for example, Batting Coach, Bowling Coach, or Fielding Coach). The Cricket Team class should have a team name, a Coach, and a group of Players.

The system should allow users to enter information about the team, its coach, and players, and then display all details along with the calculated income of each member.

The income should be calculated as follows:

- i. For a Batsman, income is the same as the base salary.
- ii. For a Bowler, income is 10% more than the base salary.
- iii. For an All-Rounder, income is 25% more than the base salary.
- iv. For a Coach, the income is the same as the base salary.

Finally, your system should also be able to compare the income of two players or two coaches to determine who earns more.