B.Sc. Engg. SWE 3rd Semester 23 December 2023 (Afternoon)

ISLAMIC UNIVERSITY OF TECHNOLOGY (IUT)

ORGANISATION OF ISLAMIC COOPERATION (OIC)

Department of Computer Science and Engineering (CSE)

SEMESTER FINAL EXAMINATION — WINTER SEMESTER, 2022-2023
DURATION: 3 HOURS — FULL MARKS: 150

SWE 4301: Object Oriented Concepts II

Programmable calculators are not allowed. Do not write anything on the question paper.

Answer **all 6 (six)** questions. Figures in the right margin indicate full marks of questions whereas corresponding CO and PO are written within parentheses.

1. a) What is dependency in object-oriented design? How does dependency on concrete classes in a code introduce coupling? Explain with an example. 8 (CO1) (PO1)

   b) You are developing a multicharacter game, wherein every character embodies a unique set of attributes and abilities. Each character is defined using a name, proficiency level (1-5), limited lifespan (0-100), distinct color (Green, Purple, Red), and a tile representing a square position within the game's landscape. All characters can move by changing their position. Each character exhibits specialized skills when in contact with another one. One character is called warrior - it specializes in attacking through strikes. It reduces the opponent's lifespan by a factor of 2 multiplied by proficiency level. Another character is called wizard - it casts spells on another character. A wizard can cast healing, poison, or rage spells. Healing restores the lifespan of another character by a factor of 2 multiplied by the proficiency level of the wizard. Similarly, poison reduces the lifespan of another character by the factor of the proficiency level of the wizard, and rage increases the strike power of another character by a factor of 1.5 of the wizard's proficiency level. 10 + 3 + 4 (CO2) (PO2)

      i. Propose an Object-oriented solution to represent the scenario using a UML diagram or your preferred programming language.

      ii. Mention any limitations to your proposed design.

      iii. State the design principles that you have used in your solution.

2. a) Any audio player has play, pause, and resume functionalities. AncientAPlayer is an audio player that can perform these functionalities by default for mp3 file type. Any advanced audio player can perform two more functionalities - skip forward and skip backward for some seconds. ProAPlayer is an advanced audio player. On the other hand, any video player can perform start, halt, and restart by default for mp4 file type. ContemporaryVPlayer is a video player that can perform these functionalities. All file types can be checked using the file name extension. Now, a person wants to play mp3 files using any video player to play only audio. 10 (CO2) (PO2)

   Which design pattern is applicable in this scenario? Draw a UML diagram or write the solution in your preferred programming language.

   b) Differentiate the following concepts with examples: 3 × 5 (CO1) (PO1)

      i. Mutable and Immutable Object

      ii. Encapsulation and Information Hiding

      iii. Annotation and Meta-Annotation

3. a) Define an annotation type for an enhancement request and apply it in a different code. Usually, a request has an ID, a synopsis of the change, and the developer's email who performed the change. The annotation must satisfy the following constraints: (
(I

- The default value for the developer's email is 'unassigned'.
- "This annotation is used to track the changes" as JavaDoc information.
- Meta information can be accessed during runtime.
- It can be used for any class, method, or attribute.

b) How do clean code guidelines complement the design principles of OOC? Write four practices of clean code and relate those with principles. 6 (CO1) (PO1)

c) Suppose, you have a collection of strings containing "level", "hello", "radar", "world", "madam", and "java" as values. 6 + 4 (CO1) (PO1)

i. Define a functional interface. Use your defined interface to count the number of palindromes in the collection.

ii. How can stream API be used to get the length of each element?

4. a) Code Snippet 1 represents a smelly code for an account class.

```
public class Account {
   String type;
   String accountNumber;
   private int amount;
   private Person accountholder;

   public void debit(int debit) throws Exception{
      if(amount <= 500)
        throw new Exception("Mininum must be > 500");
      amount = amount-debit;
      System.out.println("Now amount is " + amount);
   }

   public void transfer(Account from,Account to,int cerditAmount)
      throws Exception{
      if(from.amount <= 500)
        throw new Exception("Mininum must be > 500");
      to.amount = amount+cerditAmount;
   }
}
```

Code Snippet 1: `Account` class for Question 4.a)

i. Identify two code smells and write the refactored code. 6 (CO2) (PO1)

ii. Write a method that will print all parameters of the transfer method using reflection. 7 (CO2) (PO1)

iii. "`Account` object without the `amount` attribute needs to be stored permanently." - How would you achieve this feature? 6 (CO2) (PO2)

b) Write a generic method to find the maximal element in the specific range [from, to] of a list. Ensure proper validation for the correctness of the program. 6 (CO2) (PO1)

Consider a scenario where an online food ordering system processes orders for different cuisines, handles receipt generation, and oversees the delivery of food products based on customer preferences. `FoodOrderProcessor` class (as shown in Code Snippet 2) simulates processing food orders for different cuisines (Italian and Asian) for customers and generating receipts either through email or in physical form. You can assume that the code has no compile time error.

```
class FoodOrderProcessor {
    int cuisineType = 1; ✓
    boolean printEmailReceipt = true; ✓

    void processOrder(Customer customer, String foodItem) {
        Receipt receipt;
        if (cuisineType == 1) {
            receipt = processItalianOrder(customer, foodItem);
        } else {
            receipt = processAsianOrder(customer, foodItem);
        }

        if (printEmailReceipt) {
            sendEmailReceipt(customer, receipt);
        } else {
            printPhysicalReceipt(customer, receipt);
        }
    }

    Receipt processItalianOrder(Customer customer, String foodItem) {
        return new Receipt("Italian", foodItem, customer);
    }
    Receipt processAsianOrder(Customer customer, String foodItem) {
        return new Receipt("Asian", foodItem, customer);
    }

    void sendEmailReceipt(Customer customer, Receipt receipt) {
        System.out.println("Email receipt sent to " + customer.getEmail()
            );
    }

    void printPhysicalReceipt(Customer customer, Receipt receipt) {
        System.out.println("Physical receipt printed for " + customer.
            getName());
    }
}
class Customer {
    private String name;
    private String email;
    // constructor, getter, setter
}
class Receipt {
    private String cuisine;
    private String foodItem;
    private Customer customer;
    // constructor, getter, setter
}
```

Code Snippet 2: Skeleton of several classes for Question 5

a) Code Snippet 2 has several design smells. Identify at least 4 design smells stating their line numbers and mentioning the reason for that smell. 8 (CO3) (PO1)

b) Write the solution that satisfies SOLID principles to remove design smells. You can use Class Diagrams or your preferred OOP language. 12 (CO3) (PO2)

c) Write two test cases to check the correctness of your program written in Question 5.b). 5 (CO2) (PO1)

6. a) Draw a diagram to represent a thread life cycle. 5 (CO1) (PO1)

b) What is Deadlock in multithreading? Provide a code fragment demonstrating only the deadlock portion. 5 (CO1) (PO1)

c) Any question paper set for the semester final examination goes through several steps to ensure its quality. First, each course teacher prepares a question paper and submits it for moderation. A pool is maintained to track received questions. Multiple moderators are assigned to to moderate the available questions. In the absence of any received questions, moderators are required to wait. At the same time, once a certain limit of received questions is reached, teachers must wait before submitting additional questions. Assume the question pool has a capacity of 4, with 3 moderators and 10 teachers. 5+10 (CO2) (PO2)

Design an object-oriented solution using a UML diagram to represent this scenario. Write the corresponding code in your preferred programming language.