B.Sc. Engg. CSE 8th Semester 11 May 2023

ISLAMIC UNIVERSITY OF TECHNOLOGY (IUT)

ORGANISATION OF ISLAMIC COOPERATION (OIC)

Department of Computer Science and Engineering (CSE)

SEMESTER FINAL EXAMINATION **SUMMER SEMESTER, 2021-2022**

DURATION: 3 HOURS **FULL MARKS: 150**

CSE 4851: Design Pattern

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) Define Design Pattern and Design Principle. Explain the statement "Classes should be open for extension and closed to modification" with proper example. 2+3 (CO1) (PO1)

   b) Indicate which design pattern you will use for each of the following cases: 5 × 2 (CO4) (PO2)
      i. Be able to replace the implementation of an interface at run time.
      ii. Decouple clients of a system X from dependencies on subsystems of X.
      iii. Provide clients with a reference to an object of type X but defer the creation of an expensive object of type X until it is needed.
      iv. Define a new operation without changing the classes of the elements on which it operates.
      v. Restore a state of an object to a previous state.

   c) Hollywood principle states that, "Don't Call Us, We'll Call You". Identify a pattern satisfying this principle. Write a code example for that pattern and explain how your code satisfies this principle. 2+6+2 (CO4) (PO2)

2. a) What are the differences between the Builder and Composite pattern? 4 (CO3) (PO1)

   b) Identify a pattern which can define a one-to-many dependency between objects so that when one object changes state, all its dependents are notified and updated automatically. Briefly explain that pattern with code. Also discuss the advantages and disadvantages of that pattern. 2+10+4 (CO4) (PO2)

   c) Explain low coupling and high cohesion with an example. 5 (CO1) (PO1)

3. a) An application contains an interface `Shape` implemented by 2 concrete shapes (Circle and Rectangle). Several composite shapes can be created by using these two concrete shapes. Composite objects can be visited through a visitor from the outside of the application with the help of a `ShapeVisitor` interface. 6+6 (CO4) (PO2)

   Write code for the above-mentioned scenario using appropriate pattern and draw the corresponding UML.

   b) Consider a class that is used to represent a `Cake`. You need a number of items like egg, milk, flour to create cake. Many of them are mandatory and some of them are optional like cherry, fruits, etc. If we are going to have overloaded constructors for different kinds of cakes, then there will be many constructors and even worse, they will accept many parameters. 3+10 (CO4) (PO2)

   Identify a pattern which can solve this problem. Write corresponding implementation using the identified pattern.

4. a) Use Composite Pattern to model the notion of a folder in Windows XP. Folders may be nested and may also contain text files and binary files. Files may be opened, closed, or drawn on the screen. Folders may also have items added and removed from them. Draw the UML diagram for the described model. 7 (CO4) (PO1)

b) Which design patten restores a state of an object to a previous state? Write a code example of restoring a previous state of an object. 2+10 (CO4) (PO1)

c) Perform a comparative analysis among Singleton, Prototype and Flyweight patterns. 6 (CO3) (PO1)

5. a) Draw a UML diagram for Mediator pattern between web services and web clients. As web services, the eBay auction house and Amazon are available. Plan functions to search for an item with a textual description and buy an item from the service that gives you the best price. 8 (CO4) (PO1)

b) Identify two design patterns which reduce memory footprint. Perform comparative analysis between them. 2+6 (CO4) (PO2)

c) Identify a pattern which decouples an abstraction from its implementation so that the two can vary independently. Explain a scenario satisfying the statement. 2+7 (CO3) (PO1)

6. a) Write short notes on – “Refused Bequest” and “Large Class”. 7 (CO2) (PO1)

b) Consider the following Code Snippets - 6 × 3 (CO2) (PO2)

```
public class Rental {
    private Movie _movie;
    Private int  _daysRented;

    public Rental (Movie movie, int daysRented) {
        _movie = movie;
        _daysRented = daysRented
    }

    public int getDaysRented() {
    return _daysRented;
    }

    public Movie getMovie() {
        return _movie;)
    }

    public double amountFor() {
        double thisAmount = 0;
        //determine amounts for each line
        switch (getMovie().getPriceCode()) {
            case Movie.REGULAR:
                thisAmount += 2;
                if (getDaysRented() > 2)
                    thisAmount += (getDaysRented() - 2) * 1.5;
                break;
            case Movie.NEW_RELEASE:
                thisAmount += getDaysRented() * 3;
                break;
            case Movie.CH1LDRENS:
                thisAmount += 1.5;
                if (getDaysRented() > 3)
                    thisAmount += (getDaysRented() - 3) * 1.5;
                 break;
        }
        return this.Amount; }
}
```

Code Snippet 1: Code Snippet for Question 6.b)

```
public class Movie {
    public static final int CHILDRENS = 2;
    public static final int REGULAR= 0;
    public static final int NEW_RELEASE = 1;

    private String _title;
    private int _priceCode

    public Movie (String title, int priceCode) {
        _title = title;
        _priceCode = priceCode;

     public int getpriceCOde() {
            return _priceCode;
        }
        public void setPriceCode(int arg) {
            _priceCode = arg;
        }
        public String getTitIe {
            return _title;
        }
}
```

Code Snippet 2: Code Snippet for Question 6.b)

Answer the following questing according to Code Snippet 1 and 2.

i. Briefly explain the terms “Code refactoring” and “Code smell”.

ii. Identify two code smells which have occurred in the codes.

iii. Refactor the code removing the smells.