ISLAMIC UNIVERSITY OF TECHNOLOGY (IUT)
ORGANISATION OF ISLAMIC COOPERATION (OIC)
Department of Computer Science and Engineering (CSE)

MID SEMESTER EXAMINATION   SUMMER SEMESTER, 2022-2023
DURATION: 1 HOUR 30 MINUTES   FULL MARKS: 75

CSE 4851: Design Patterns

Programmable calculators are not allowed. Do not write anything on the question paper.
Answer all 3 (three) questions. Figures in the right margin indicate full marks of questions with corresponding COs and POs in parentheses.

1. a) What are the main principles of OOP? Explain the Dependency Inversion Principle with an appropriate code example.
   4
   (CO1)
   (PO1)

   b) Write down the name of the design pattern or principle that would be most useful for addressing each described situation below. Explain the reason for choosing the particular pattern or principle.
   3 x 3
   (CO3)
   (PO1)

      i. You are building a system that relies on a complex algorithm, the algorithm may be changed often due to marketing pressures. What pattern would be appropriate to support this situation?

      ii. A pizza factory produces pizzas with various toppings. There are 20 different toppings and a customer may order any combination of toppings. Assume that each of pizza bread and each topping will be represented by a different class.

      iii. You are building a cricket app that notifies viewers about the information such as current score, run rate etc. Suppose we have made two display elements *CurrentScoreDisplay* and *AverageScoreDisplay*. *CricketData* has all the data (runs, bowls etc.) and whenever data changes the display elements are notified with new data and they display the latest data accordingly.

   c) Hollywood principle states 'Don't Call Us, We'll Call You'. Identify a pattern that satisfy this principle. Write a code example for that pattern and explain how your code satisfies this principle.
   10
   (CO4)
   (PO2)

2. a) Propose a scenario where the Builder pattern will be beneficial for improving readability, flexibility, and maintainability of object creational code. Provide pseudocode for object creation part only.
   5 + 5
   (CO3)
   (PO1)

   b) Imagine a situation where you have software that implements an online store that sells knives called *KnifeStore*. The software produces *SteakKnives* and *ChefsKnives*. The methods of sharpening, polishing, and packaging would remain in the order *Knife* method. However, the responsibility of creating the product will be delegated to another class: a *KnifeFactory*. A *KnifeFactory* has two subclasses named *BudgetKnifeFactory* and *BasicKnifeFactory*. *BudgetKnifeFactory* can produce *BudgetChefsKnife* and *BudgetSteakKnife*. Similarly, *BasicKnifeFactory* can produce two types of knives, *BasicChefsKnife* and *BasicSteakKnife*.
   Identify a pattern that will be appropriate for the given scenario. Write the code implementation and draw the UML diagram of your implementation.
   15
   (CO4)
   (PO2)

3. a) Which design pattern works as a bridge between two incompatible interfaces. Explain the intent and motivation of this pattern.
   2 + 7
   (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 how the identified pattern will ensure loose coupling among objects. Also discuss the advantages and disadvantages of that pattern.

6 + 6
(CO3)
(PO1)

c) Differentiate between

3 × 2
(CO3)
(PO1)

i. Builder and Factory Pattern

ii. Strategy and Decorator Pattern

iii. Interface and Abstraction Pattern