B.Sc. Engg. SWE 5th Semester 03 March 2020 (Morning)

# ISLAMIC UNIVERSITY OF TECHNOLOGY (IUT)

## ORGANISATION OF ISLAMIC COOPERATION (OIC)

## Department of Computer Science and Engineering (CSE)

**MID SEMESTER EXAMINATION** **WINTER SEMESTER, 2019-2020**

**DURATION: 1 Hour 30 Minutes** **FULL MARKS: 50**

## SWE 4501: Design Patterns

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

There are **3 (three)** questions. Answer **all** of them.

Figures in the right margin indicate marks.

**This is an Open Book exam.**

The examinees can keep **one copy** of book or other printed materials with them, however, they cannot share the materials with other examinees. The material must be in book binding or spiral binding.

For all three questions, consider that you are developing a game "save the animals" where the main character, Ratul, fights the bad guys to free animals they have illegally captured. When answering the questions, consider the followings:

- Use standard symbols when drawing a class diagram. Underline a method that is abstract.
- If your solution uses a factory method or a template method design patterns, write pseudo code for the factory/template method even if the question asks only a class diagram.
- Use meaningful names of items in both code and class diagram instead of random characters like "A", "X", "M" etc.
- When it is asked to write code, you can write pseudo code instead of code in a particular language.

1. Ratul has two actions: close action and distant action. 20

The gamer can choose from one of three close actions for Ratul: *punch*, *kick* and *head*.

The distant action is a sequence of acts: *move*, *grab* and *hit*. The gamer can choose from one of two distant actions: *power* and *skill*.

When using *power* action, Ratul moves to a nearby enemy, picks the enemy up and slams the enemy to the ground. When using the *skill* action, Ratul moves to a nearby enemy, holds the enemy's collar, and hit the enemy's belly with a knee. Note that the *move* part is exactly the same for the two actions, but the *grab* and *hit* parts vary.

Draw a class diagram using types, methods and fields from the scenario above.

2. Gamers' statistics are usually stored in a plain text file. However, some users prefer to keep their statistics secret so they want to encrypt the file. For some, the data is very large so a compression facility is also required. Additionally, Some people prefer to keep their data as base-64 encoding. 17

These conversions of the data file can be done in any sequence and the same conversion can be applied multiple times.

Write code for the scenario above.

Also, demonstrate the usage of your code for the following cases:

- A data file is first compressed then encrypted.
- A data file is encrypted, then compressed, then encoded, then compressed again.
- No conversion is done to the data file.

3. The users of "save the animals" can save their preferences. To implement this, you need to write a `Preference` class with the following considerations: 13

   - There should be a `void setPreference(string key, string value)` method to store a preference.
   - There should be a `string getPreference(string key)` method to retrieve a preference.
   - A single instance of the `Preference` class should be maintained throughout the application so that all the code gets the same preference values.
   - Make sure that no code can accidentally create additional instances of `Preference` class.
   - The instance of the `Preference` class should be created only when it is needed for the first time.
   - Computer games are usually multi-threaded programs.

   Write the `Preference` class.

   Write a method `demo()` outside the `Preference` class to demonstrate the usage of `setPreference()` and `getPreference()` methods. Your `demo()` method should not contain more than five statements.