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

SEMESTER FINAL EXAMINATION   **WINTER SEMESTER, 2020-2021**
DURATION: 1 Hour 30 Minutes   **FULL MARKS: 75**

CSE 4301: Object Oriented Programming
Answer all **three (3)** questions.
Figures in the right margin indicate marks along with COs and POs.

1. Assume that you are asked to develop a part of a program that stores the information of a generic   (30)
    hospital that follows the following scenarios. You can add necessary member functions and variables   (CO4)
    that are not mentioned explicitly in the scenarios.

    In a hospital, there are a number of **physicians** and **patients**. Assume that, there are **three** categories of
    physicians: emergency, indoor, outdoor. Each **admitted** patient is under the supervision of one **indoor
    physician**. On the other hand, one **indoor physician** can supervise multiple patients but each physician
    has a different capacity of maximum_no_of_supervision.

    Create class definitions which serve the above requirements. More about the class attribute and
    behavior is mentioned below.

    All patients have a **name**. Each **new** patient in the hospital will get a **unique patientID**. Similarly, all
    physicians have a **name** and a **unique ID**. As well as they have an attribute called **speciality**.

    Hospital class has an attribute called **name** which stores the name of the hospital.

    Write a **public** member function of **Hospital** following prototype **bool admit(Patient & p, Indoor & I);**
    it assigns the Patient object p under the supervision of indoor physician I. The **admit** function will
    check whether he/she can supervise this patient or not (**Remember** there is a limit on the highest
    number of patients, one indoor physician can supervise at any moment). If it fails to assign the patient,
    admit() will return **false**, otherwise it returns **true**.

    Write a **public** member function of the **Hospital** following prototype **bool discharge(Patient & p);** it
    discharges (removes) the Patient object p from the assigned physician (assigned by admit function) of
    the hospital. If a patient is not found then the discharge function will return **false**, otherwise it returns
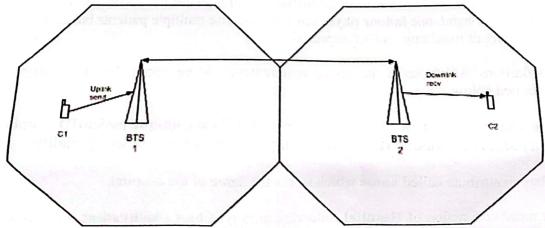   **true.**

    Write a **public** member function of the **Hospital** following prototype **void showStatus(void);** it will
    print hospital name with total count of patients and doctors present as well as all the Indoor Physicians'
    information with their assigned patients. **Sample output** is in the following for a hospital "**IUT
    Medical Center**" which has 2 doctors and 4 patients currently.

    The expected output will be -

2. In a cellular network system, there are **BaseStation (BTS)** and **cellular devices (C)**. A number of C objects registered under one **BTS**. A cellular object does not communicate directly to other cellular objects. It communicates via a **BTS**. The sending cellular object sends data to its registered **BTS** (Uplink Communication). Then that **BTS** will send the data to the **BTS** where the receiver cellular object is registered to. After that, the receiver cellular object receives data from the **BTS** (Downlink Communication). (30) (CO4)

Each **BTS** contains a number of **resource block** objects (**RB**) for both uplink and downlink cellular communication. Uplink and Downlink resources are different. Assume that **BTS** assigns one **uplink** and one **downlink** RB object to each registered C when it starts any cellular communication. After communication ends, BTS yields back the resources so that it might be used by another C starting cellular communication. It should be noted that, a C object registered under one BTS does not mean that C has started cellular communication.

Write an OOP program where you can simulate the above scenario. To simulate you need class definitions of BTS, C and RB. In the following table some noteworthy attributes and behaviors are mentioned. You must incorporate these attributes and behaviors. You may need to add functions other than these. You are not required to write down the main().

You need to write exception classes also to throw specific exceptions. In the following table scenarios are mentioned where the exceptions should be raised.

Class Name	Attributes	Behavior	Exception
BTS	2D Coordinate, uniqueID	**allocateRBCell(RB*,C*)** **deallocateRBCell(RB*,C*)** **getFreeRBs()** - *return a vector of free RBs under that BTS*	1. Trying to allocate already allocated RBs. 2. Trying to allocate another RB to a Cellular Device which is already using RB. 3. A cellular device want to start the communication but no free RBs are present
C	2D Coordinate, currentTransmissionPower, minTransmissionPower, maxTransmissionPower, uniqueID	**startCommunication()** **endCommunication()** **status()** - *this function will print the uniqueID of C. If it is under cellular communication then it shows the uniqueID of RBs of uplink and downlink.*	
RB	currentTransmissionPower, minTransmissionPower, maxTransmissionPower, isUplinkRB, channelFrequency, uniqueID	**status()** - *this function will print the uniqueID of RB. If downlink RB then print* minTransmissionPower, maxTransmissionPower	

3. Write member function to read and write to file for the class definitions of question 1. So that when we restart, the program can be resumed from the last place where it is closed. (15) (CO4)