NATIONAL UNIVERSITY
OF COMPUTER & EMERGING SCIENCES
PESHAWAR CAMPUS

Problem Set:	Assignment: A04	**Semester**:	Spring 2019
Points:	*10*		
Date Set:	*See SLATE*	**Due Date**:	*See SLATE*
Course:	CS217 - OOP	**Instructor**:	Dr. Nauman

1 Class Design Case Study

In this assignment, we are going to look at a case study which will help us understand the concepts related to class design.

Class design the idea of looking at a real world problem and figuring out which classes should be created in code to represent real world entities. It also covers further details such as which member fields should be in each class (based on the real world requirements) and which methods will be required to carry out the operations. Further, a good class design uses the concepts of OOP – such as inheritance, encapsulation, data hiding and polymorphism – as and when needed.

In the following section, you will be given a detailed (plain text) description of a housing website. A company wants to have this website developed and you are asked to create a class design based on the given requirements.

Note that you **do not have to write the definitions** of any of the methods – you just have to write the class declarations (including fields and methods). However, do make sure that you pay close attention to the concept of abstraction and only make those fields and methods accessible from outside the class which are absolutely necessary. Everything else should be abstracted away. Also, make sure you set the proper types for each of the fields and methods.

2 A Housing Website

The concept of a housing sale and rental website is fairly simple: different users can sign up to the site – some of them will be buyers, some will be sellers and a few will be website administrators. Each user is assigned a unique identifier. They should also be allowed to set a password for their account. We would also need to automatically generate a unique identifier for new users (which would increase sequentially as we get more users).

Particularly, sellers post some images and text description of a property they wish to sell or rent out. Images would be files which are stored on the server and we need to keep track of their relative paths as well as their size. We need to keep track of which properties a seller has (including those that have already been sold). We also wish to keep track of the earnings of the seller. They also need to be verified to reduce the risk of fraud.

Buyers, on the other hand, can add credit to their account which would allow them to purchase properties.

Properties are a little more detailed since they are the primary entity in our system. A property can be of several types – apartment, house, condo and a single room. Each property should have several pieces of information associated with it: area of the property, the status (sold or available), cost, images and a detailed description.

Each type of property has some specific pieces of information too. Apartments, houses and condos have number of rooms and whether they are furnished or not. Rooms (obviously) do not have number of rooms but they need to specify whether the room is shared and how many people can share the room. Condos also need to specify whether there are amenities associated with them such as pool, gym and play areas.

A property can be made available by a seller and a buyer should be allow to buy the property once it's available – changing the status of the property from available to sold.

From a technical perspective, we need to create a database manager that will handle details of connecting and disconnecting from a database. It should also store the authentication information such as database username, password and server location. However, we want to ensure that we can support multiple types of databases so we also need specific types of database managers – one to handle MariaDB and another to handle Postgres (both of which are very popular databases). The logic for connecting to both of these will be significantly different so the class design needs to take this into consideration.

Finally, we need to have a rating system which allows buyers to rate a property. For this, we need to keep track of which buyer is rating which property and the actual rating (in number of stars from 1 to 5).

3 Task

Create a class design based on the above case study. Note again that you do not have to provide definitions of the class methods. So, this will not be a working website. You are only required to create the class declarations (including everything associated with it) based on the concepts of OOP studied so far. You do have to ensure though that the declarations are in proper C++ syntax and follow good coding practices.

4 Submission

After you're done with your class design, write down (**read: assignment must be hand-written**) the full code you've created and submit that on or before the deadline. Late assignments (or soft copies) will not be accepted.