**National University of Computer & Emerging Sciences, Karachi**
**Spring-2020 CS-Department**
**Final Examination**
**23RD June 2020, 09:00 a.m. – 12:30 p.m.**

| | |
|---|---|
| **Course Code: CL-217** | **Course : Object Oriented Programming Lab** |
| **Instructor Name: Muhammad Irfan Ayub** | |
| **Student Roll No:** | **Section:** |

**INSTRUCTIONS:**

- Read the questions carefully.
- There are **02** questions on **04** pages in this exam, each of **25** points.
- You may take an assumption but it should not contradict with the actual logic/concept/meaning of the question.
- You **MUST** follow the submission guidelines provided.

**SUBMISSION GUIDLINE**

- Name all your code files appropriately (using classes/main names).
- Name of the output screenshot should be in the format **"Q01-18K0000".** The screenshot **MUST** include the output command with the source file path (where your program is stored).
- Put all the files (including the output screenshots) in a folder named as your ID in the format **"K180000"**.
- Zip the folder with the same name (**.zip file**).
- Upload and Submit the file before deadline.
- You have to submit by both, **GOOGLE CLASSROOM & EMAIL (muhammad.irfan@nu.edu.pk)**
- **Late submission IS NOT acceptable under any circumstances.**

**Time**: 180 minutes + 30 Minutes for submission.
**Max Marks:** 50 points

**Question - 01:** **25-Points**

When a secret code is shared over a transmission channel, it is usually sent as a sequence of bits, that is, 0s and 1s. Due to noise in the transmission channel, the transmitted code may become corrupted. That is, the code received at the destination is not the same as the code transmitted; some of the bits may have been changed. There are several techniques to check the validity of the transmitted code at the destination. One technique is to transmit the same code twice. At the destination, both copies of the code are compared bit by bit. If the corresponding bits are the same, the code received is error-free.
You are required to write a program to check whether the code received at the destination is error-free. For simplicity, assume that the secret code representing the code is a sequence

of digits (0 to 9) and the maximum length of the code is 250 digits. Also, the first number in the code is the length of the code.
For example, if the secret code is:

**7 9 2 7 8 3 5 6**

then the actual code is 7 digits long, and it is transmitted twice.
The above code is transmitted as:

**7 9 2 7 8 3 5 6 7 9 2 7 8 3 5 6**

**Input:** A file containing the secret code and its copy
**Output:** The secret code, its copy, and a code—if the received code is error-free—in the following form:

| Code Digit | Code Digit Copy |
|---|---|
| 9 | 9 |
| 2 | 2 |
| 7 | 7 |
| 8 | 8 |
| 3 | 3 |
| 5 | 5 |
| 6 | 6 |

Code transmitted OK.

**Question - 02:** **25-Points**

An Educational Organization offers various types of user accounts, such as students, teachers, staff with their specific needs. Two of the most commonly used accounts are student and teacher. Each of these accounts has various options. For example, a student account that requires a maximum course load and a minimum CGPA for not falling into the warning. Similarly, a teacher account that limits the number of courses you can teach. Another type of account is a staff account. In this programming task, you use abstract classes and pure virtual functions to design classes to manipulate various types of accounts. For simplicity, assume that the educational organization offers three types of accounts: student, faculty, and staff, as described next.

**Students accounts:** Suppose that the educational organization offers two types of student accounts: one that has a higher maximum course load and a lower CGPA requirement for warning (under graduate students) and another that requires a lower maximum course load and has a higher CGPA requirement for warning (graduate students).

**Faculty accounts:** Suppose that the educational organization offers three types of faculty accounts: one with a monthly salary, medical allowance, provident fund, and paid leaves; another with no monthly salary, lower medical allowance, lower provident fund and lesser paid leaves; and a third with no monthly salary, no medical, no provident fund, and no paid leaves.

**Staff account:** In an account of this type, employment is entirely on contractual basis and these accounts have lower medical allowance, no provident fund and lesser paid leaves. For training purposes, the staff is also taking courses to enhance their skills.

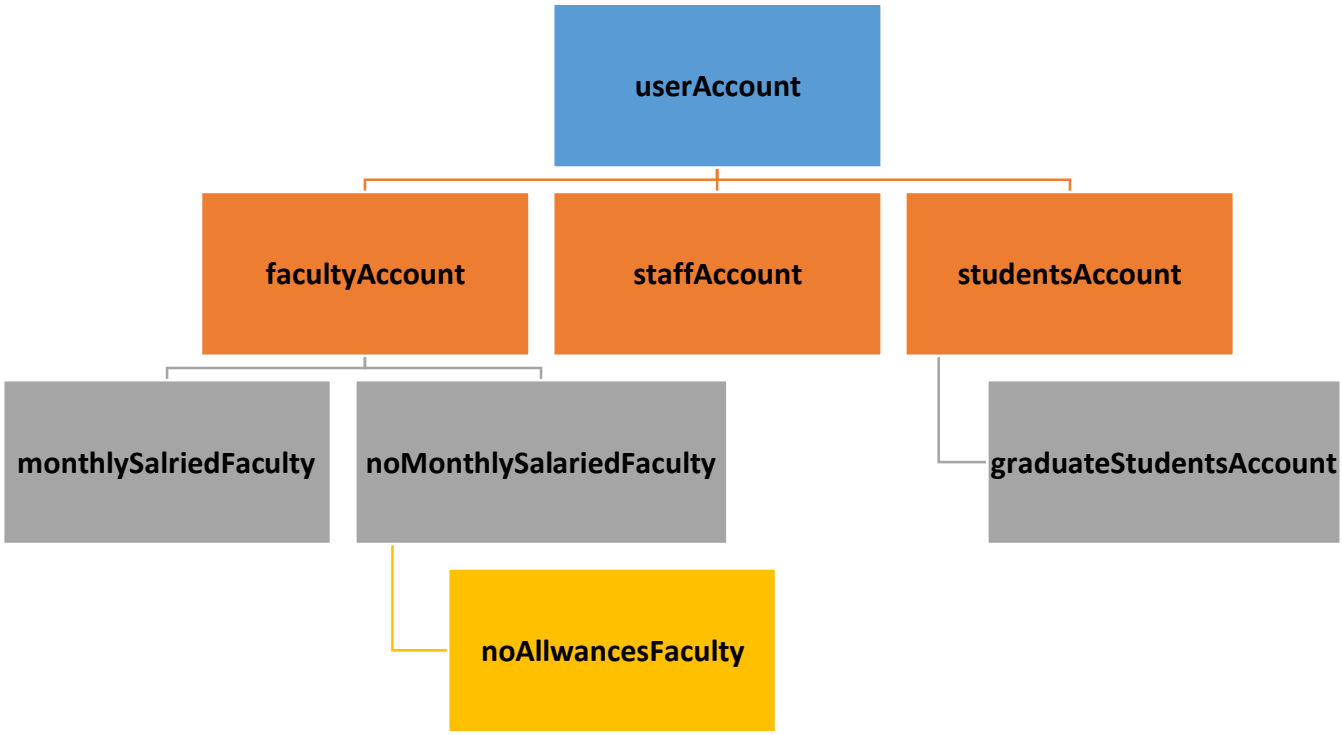


Figure 1 shows the inheritance hierarchy of these accounts.

Note that the classes **userAccount** and **facultyAccount** are abstract. That is, we cannot instantiate objects of these classes. The other classes in Figure 1 are not abstract.

userAccount: Every user account has an ID, the name of the user, course load. Therefore, instance variables such as id, name, courseLoad should be declared in the abstract class userAccount. Some operations common to all types of accounts are retrieve account user's name, user id, and course load; add course; withdraw course; and check monthly progress. So include functions to implement these operations. Some of these functions will be pure virtual.

facultyAccount: A faculty account is a user account. Therefore, it inherits all the properties of a user account. Because one of the objectives of a faculty account is to be able to give grades, include the pure virtual function giveGrade to give a grade.

monthlySalariedFaculty: A monthly salaried faculty account is a faculty account. Therefore, it inherits all the properties of a faculty account. For simplicity, assume that

this type of faculty has a monthly salary, medical allowance, provident fund, and paid leaves Include appropriate named constants, instance variables, and functions in this class.

noMonthlySalariedFaculty: A faculty account with no monthly salary is a faculty account. Therefore, it inherits all the properties of a faculty account. Furthermore, this type of account has no monthly salary, lower medical allowance, lower provident fund and lesser paid leaves;

noAllowncesFaculty: A faculty account with no allowances is a faculty account with no monthly salary. Therefore, it inherits all the properties of a no monthly salaried faculty account. Furthermore, this type of account has no monthly salary, no medical, no provident fund, and no paid leaves.

studentsAccount: A student account is a user account. Therefore, it inherits all the properties of a user account. Furthermore, a student account also has a CGPA.

graduateStudentsAccount: A graduate student account is a student account. Therefore, it inherits all the properties of students account. It has a higher minimum CGPA requirement and lower maximum course load.

staffAccount: A staff account is a user account. Therefore, it inherits all the properties of a user account. In addition, it has instance variables to store the number of hour worked for current month, hourly rate, and total served time.

Write the definitions of the classes described in this programming task and a program to test your classes.