BACS2023 Object-Oriented Programming
Practical #6A

Question 1

A. A bank offers its customers the following account types:
   - The *savings account* earns interest that compounds monthly i.e., the interest is calculated based on the balance on the last day of the month.
   - The *current account* has no interest, but the customer is given a small number of free transactions per month and is charged a nominal transaction fee for each additional transaction.

   Create a superclass **Account** that has the following properties:
   - ✔ *Account number*, *balance*, and *date created*
   - ✔ Methods for *deposit* and *withdrawal*

   Create two subclasses for savings and current accounts. The class for savings account should have methods to calculate the interest amount and to add the interest to the balance.

B. Write a test program that creates objects of **SavingsAccount** and **CurrentAccount**. Test the deposit and withdraw methods in each class to ensure that they work correctly. For the **SavingsAccount** class, include statements to test the methods for calculating the interest amount and adding the interest to the balance.

C. Write a client program that has a simple menu for handling transactions for a current account. Your program should ensure that the transaction fee is deducted from the account balance if the number of free transactions has been exceeded for the month.

Question 2

A. Override the **withdraw()** and **deposit()** methods in the **CurrentAccount** class such that the transaction fee is automatically deducted for each transaction once the number of free transactions has been exceeded. Next, simplify your client program accordingly.

B. Write the **toString()** method for the **Account** class such that it returns the object's data field values. Next, override the **toString()** method in the **CurrentAccount** class so that the object's transaction count is also returned as part of the string. Modify your test program from Question 1B to test the **toString()** methods.

C. Override the **equals()** method in the **Account** class such that it returns **true** if the current object has the same account number as the parameter. Test to ensure that your method works correctly.

Question 3

OOP and procedural programming are two different programming paradigms. Table 1 below shows how a problem is represented by the two paradigms. Identify the type of programming paradigms represented by View A and View B. Which paradigm resembles the real-world better? Explain why.

Table 1: Different views of two programming paradigms

View A	View B
Area ➢ of Cube Volume ➢ of Cube	Cube ➢ Area ➢ Volume