Department of Statistics & Computer Science

University of Kelaniya
Academic Year – 2022/2023
COSC 12043 / BECS 12243 - Object Oriented Programming
Tutorial 08

1. Write a method called **calculateGrade** that takes an integer score (0-100) as input from the user and returns a grade based on the following criteria:

   A for 90-100

   B for 80-89

   C for below 70

   If the user enters a value outside the range of 0-100, display an error message.

   ```
   Enter your score (0-100): -4
   Error: Please enter a score between 0 and 100.

   Enter your score (0-100): 80
   Your grade is: B
   ```

2. Write a Java program that prompts the user to enter the number of elements in an array and then input the elements of the array. The program should call a method named **minGap**, which accepts the array as a parameter and calculates the minimum gap between adjacent elements, defined as the absolute difference between consecutive elements. If the array contains fewer than 2 elements, the method should return 0. The main method should display the result returned by **minGap**. For example, if the user enters the array {1, 3, 6, 5, 12}, the gaps are 2 (3 - 1), 3 (6 - 3), 1 (6 - 5), and 7 (12 - 5), so minGap should return 1.

3. Write a method called **allLess** that accepts two arrays of integers and returns true if each element in the first array is less than the element at the same index in the second array. Your method should return false if the arrays are not the same length.
   For example, if the two arrays are [10, 20, 30, 40, 50] and [35, 30, 60, 73, 55], it should return true after a call to your method.

4. Create a program to manage product details in a shopping system. Each product should have attributes such as name, price, and discount percentage. Implement a method to initialize these details and calculate the final price after applying the discount.

5. Write a Java class called **BankAccount**, which has two fields name (String) and balance (double), two constructors, and five methods getName(), getBalance(), deposit(double amount), withdraw(double amount) and displayInfo(). The first constructor should initialize the name to null and balance to 0. The second constructor initializes the name and balance to the parameters passed. deposit method deposits the amount to the account causing the current balance to increase, the withdraw method withdraws the amount causing the current balance to decrease and the displayInfo() method returns the account name and the current balance separated by a comma.

   For example, if the name is Saman and the balance is 40000.0 it should return:

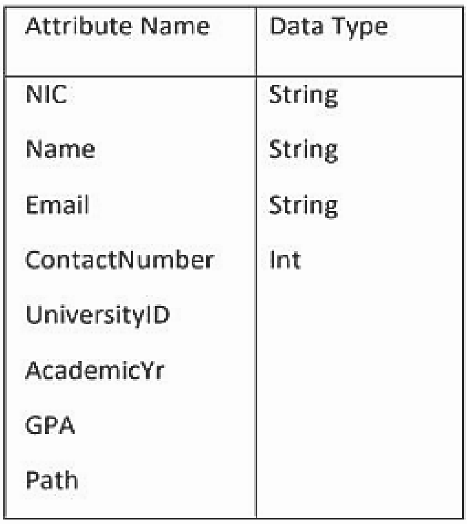
   ```
   Saman, Rs.40000
   ```

   Write a client program called **BankAccountClient** that creates a BankAccount object called B1 and assigns "Peter" to its name field and 10000 to the balance field. Call the deposit method to deposit 8000 to this account and call the getBalance() method to display the current balance. Then, withdraw 3000 from this account and print the object using println. Create another object called B2 without passing parameters and display the name and current balance for this object.

   a. Create the above two classes (BankAccount and BankAccountClient) in one Java file. Can you make both classes public? Justify.
   b. Create the above two classes in two Java files and re-run the program.

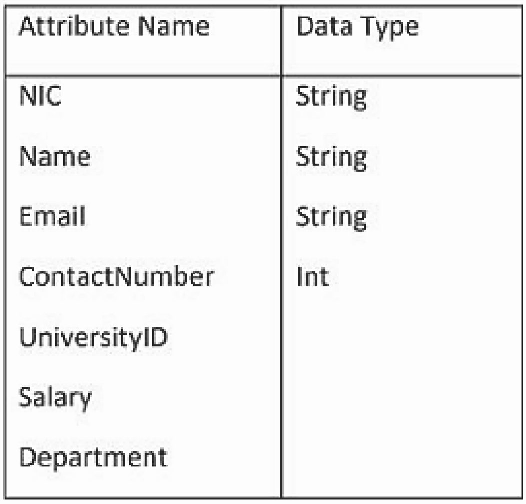
6. Consider digitizing the Faculty Information System (FIS) of the Faculty of Science, University of Kelaniya. The users who are involved in the system is the **Student** and **Staff**.

   Student consists of the following details:



Attribute Name	Data Type
NIC	String
Name	String
Email	String
ContactNumber	Int
UniversityID	
AcademicYr	
GPA	
Path	

   Staff consists of the following details:



Attribute Name	Data Type
NIC	String
Name	String
Email	String
ContactNumber	Int
UniversityID	
Salary	
Department	

a. Create Student and Staff Classes. Each Class would be coded in separate Java files.

b. Write the following methods to output the attributes that are stored,

   For Student class,

   i. displayInfo()
   ii. GpaIncrease() to increment the GPA of second-year undergraduates by 0.0001.

   For Staff class,

   i. displaystaffInfo()
   ii. sallaryIncrement() which increment the salary for the staff who belong to the Computer Science Department by 5%

c. In this situation, Dr. Saman Perera is a lecturer from the Computer Science Department who holds the UniversityID U001 and NIC of 1111, Ms. Nilanka Silva is a second-year student who holds the UniversityID PS/2016 /001 and NIC of 9999. Create the objects from the two classes and store the above details and call above defined methods for both objects.

7. Write a program to print the area and perimeter of a triangle having sides of 3, 4, and 5 units by creating a class named **Triangle** without any parameter in its constructor.

8. Write a program to print the area and perimeter of a triangle with sides of 3, 4, and 5 units by creating a class named **Triangle** with a constructor with the three sides as its parameters.

9. Write a program to print the names of students by creating a **Student** class. If no name is passed while creating an object in the **Student** class, then the name should be "Unknown", otherwise the name should be equal to the String value passed while creating the object of the Student class. (Use Constructor appropriately)

10. Suppose you have a bank account with an initial amount of Rs. 0 and you have to set the amount when you create bank account objects. Create a class **BankAccount** with a data member named amount with an initial value of RS. 0. Now make two constructors of this class as follows:

    - Without any parameter - no amount will be added to the bank
    - Having a parameter which is the amount that will be added to the **BankAccount**

    Create objects of the **BankAccount** class and display the amounts in it.

11. Consider the below-given code:

```
class Counter{
      int count=0;      //instance variable
      Counter(){
            count++;    //incrementing value
            System.out.println(count);
      }
      public static void main(String args[]){
            Counter c1=new Counter();
            Counter c2=new Counter();
            Counter c3=new Counter();
      }
}
```

a. What will be the output?

b. If the count variable is changed as

```
      static int count = 0;
```

will the output be changed? Give reasons.