Student ID:	
Machine No:	

Sri Lanka Institute of Information Technology

B.Sc. Honours Degree in Information Technology

Specialized in Information Technology

Final Examination (Computer Base)
Year 2, Semester 1 (2023)
Paper Version B

IT2030 – Object Oriented Programming

Duration: 3 Hours

June 2023

Instructions to Candidates:

- This paper contains four questions. Answer All Questions.
- Marks for each question given on the paper. Total Marks: 100.
- Create a folder on the Desktop with your student registration number and store all your program files inside that.
- Create a separate Project for each question. The name of the project is provided.
- This paper contains 07 pages including the Cover Page.

Instructions to Candidates when submitting:

- Save all your work.
- Delete all unnecessary files from each project folder (There should be 4 folders name as Question01, Question02, Question03 and Question04 inside your ID folder, and each folder should contain the answer (.JAVA files ONLY)).
- Zip the Student ID folder (Zipped folder also should be named with Student ID number).
- Upload the zipped folder into the correct link.

Question 1 (20 marks)

This question is based on the **OOP concepts**.

Implement the necessary classes, along with their required attributes and methods, based on the following description.

You have been assigned the task of developing a Student Information Management System for a school. The system should consist of two classes, **Student** and **Course**.

The Student class should have several data members, including ***StudentId, name, address***, and ***contact number***, all of which are of string data type. It should also have a constructor to initialize these attributes and two methods, **Read()** and **Print()**. The **Read()** method should accept inputs for the above-mentioned attributes through the keyboard, and the **Print()** method should display their values.

The **Course** class should have data members such as ***CourseId, name***, and ***instructor***, all of which are of string data type. It should have a constructor to initialize these attributes and a method, **Read(),** to input the values for these attributes by calling the **Read()** method of the Student class. The **Course** class should also have a method called **AddStudent()** that allows adding a student to the course. This method should take a Student object as a parameter and add it to an array list of enrolled students.

Furthermore, you need to create a class called **StudentApp** with a main method. In the main method, you should create 2 objects each of the **Student** and **Course** classes, input their details using the **Read()** method, and add students to courses using the **AddStudent()** method. You should store Student objects in an ArrayList, ensuring that only objects of the **Student** class and its subclasses can be added and store Course objects in an ArrayList, ensuring that only objects of the **Course** class and its subclasses can be added to that.

Finally, you should display all the objects stored in the ArrayList by calling each object's **Print()** method. This will show the details of the students and the courses they are enrolled in.

Hint: Refer to the sample output given below.

Save the project as **Question01**

Sample Output:

```
Student Information:
Student ID is : IT111
Student Name is : Kamal Perera
Student Adress is : Colombo 2
Student Telephone is : 0771111111

Student ID is : IT222
Student Name is : Nimali Herath
Student Adress is : Colombo 3
Student Telephone is : 0773333333

Course Information:
Course ID: CID123
Course Name: Object Oriented Programming
Instructor: Sumali Perera
Enrolled Students:
Kamal Perera
Nimali Herath

Course ID: CID456
Course Name: Database Management Systems
Instructor: Priyal Silva
Enrolled Students:
Nimali Herath
```

Question 2 (25 marks)

This question is based on the **Collection Framework and Generics.**

a) Write a program that takes input from the user and creates an ArrayList of integers. The program should then remove all the even numbers from the ArrayList and display only the odd numbers in the ArrayList.

i) Create an empty ArrayList of **integers** using the ArrayList class. Then prompt the user to enter numbers, one at a time, until they enter 0. Each number the user enters must be added to the ArrayList. [6 marks]

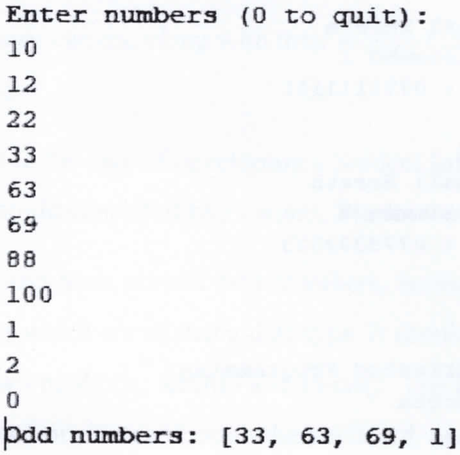
ii) Once the user has entered all the numbers, remove any even numbers from the ArrayList. [6 marks]

iii) Display the remaining odd numbers in the ArrayList [3 marks]

Hint: Refer to the sample output given below.

Save the project as **Question02A**

Sample Output:



```
Enter numbers (0 to quit):
10
12
22
33
63
69
88
100
1
2
0
Odd numbers: [33, 63, 69, 1]
```

b) Write a generic class called **"Pair<T, U>"** that represents a pair of two values, one of type **T** and another of type **U**. The class should have the following methods.

[1 mark]

i) Write a parameterized constructor that initializes the first and second values of the pair. [2 marks]

ii) Write a method called **"getFirst()"** which returns the first value of the pair. [2 marks]

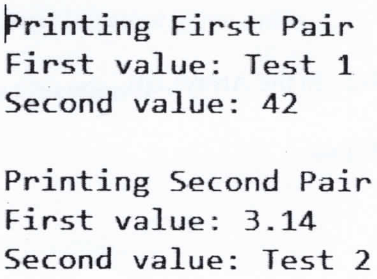
iii) Write a method called **"getSecond()"** which returns the second value of the pair. [2 marks]

iv) Write a class called **"MainApp"** which contains the main method. Create two objects where the first object accepts a **String** as the first value, **Integer** as the second value of the pair. The second object should accept a **Double** as the first value and a **String** as the second value. [3 marks]

Hint: Refer to the sample output given below.

Save the project as **Question02B**

Sample Output:



```
Printing First Pair
First value: Test 1
Second value: 42

Printing Second Pair
First value: 3.14
Second value: Test 2
```

Question 3 (15 marks)

This question is based on the **Threads implementation**.

You are tasked to write a program to print odd and even numbers using two threads. Refer to the partially completed **EvenOddPrinter** class with the main method below.

```
public class EvenOddPrinter {
    private static final int MAX_NUM = 20;
    private static volatile int nextNum = 1;
    private static final Object lock = new Object();

    public static void main(String[] args) {
      Thread evenThread = new Thread(new EvenRunnable(), "EvenThread");
      Thread oddThread = new Thread(new OddRunnable(), "OddThread");

      //Complete the rest of the code here
  }
```

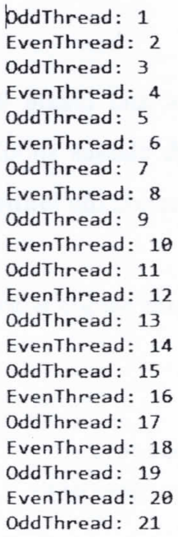
a) Create a static class called "**EvenRunnable**" which can generate even numbers from 1 to **MAX_NUM** in order. [6 marks]

b) Create another static class "**OddRunnable**" which can generate odd numbers from 1 to **MAX_NUM** in order. [6 marks]

c) Use synchronization to make sure that odd and even numbers are displayed in order as shown in the sample output below. [3 marks]

Save the project as **Question03**

Sample Output:



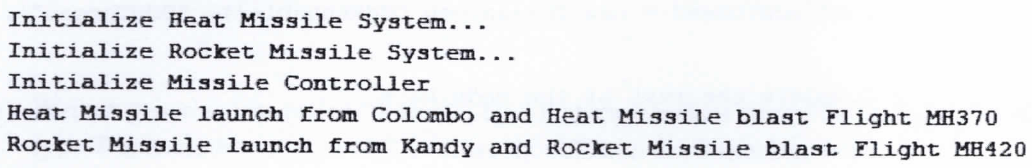
```
OddThread: 1
EvenThread: 2
OddThread: 3
EvenThread: 4
OddThread: 5
EvenThread: 6
OddThread: 7
EvenThread: 8
OddThread: 9
EvenThread: 10
OddThread: 11
EvenThread: 12
OddThread: 13
EvenThread: 14
OddThread: 15
EvenThread: 16
OddThread: 17
EvenThread: 18
OddThread: 19
EvenThread: 20
OddThread: 21
```

Question 4 **(40 marks)**

This question is based on the **Design Patterns** implementation.

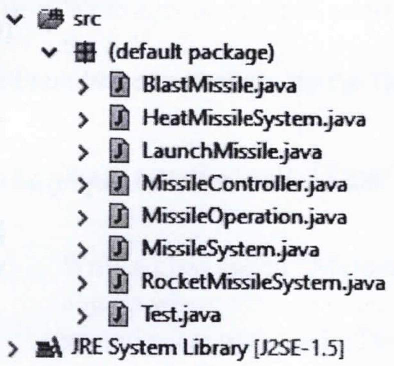
You have been asked to implement a **WAR System** for launching missiles. Assume that there are two types of missile systems (Heat missile and Rocket missile) you can launch and blast them once the source and destination locations are provided. There is a one controlling point called Missile Controller is used to set blast or launch operations for each missile type. Missile systems and controllers are created according to the **Singleton pattern** and overall application developed according to the **Command pattern**. Please refer the below screenshot of console outputs and your program also should display the same output.

Console Output:



```
Initialize Heat Missile System...
Initialize Rocket Missile System...
Initialize Missile Controller
Heat Missile launch from Colombo and Heat Missile blast Flight MH370
Rocket Missile launch from Kandy and Rocket Missile blast Flight MH420
```

Package Hierarchy:



```
v src
    v (default package)
        > BlastMissile.java
        > HeatMissileSystem.java
        > LaunchMissile.java
        > MissileController.java
        > MissileOperation.java
        > MissileSystem.java
        > RocketMissileSystem.java
        > Test.java
> JRE System Library [J2SE-1.5]
```

a) Implement the **MissileSystem** interface as given below and create 02 classes named **HeatMissileSystem** and **RocketMissileSystem** and both classes should implement the **MissileSystem** interface and in each class, you should override the **launch** (String source) and **blast** (String destination) methods. [20 marks]

[Hint: - HeatMissileSystem and RocketMissileSystem classes should work according to the singleton pattern]

```
public interface MissileSystem {
            public void launch(String source);
            public void blast(String destination);
}
```

b) Refer the **MissileOperation** interface as given below and declare the **initiateOperation(String location)** method. Now implement this interface in **BlastMissile** and **LaunchMissile** classes and override the **initiateOperation(String location)** method in each class. [08 marks]

```
public interface MissileOperation {
        public void initiateOperation(String location);
}
```

c) Implement **MissileController** class that set two operations called launch and blast. Then you will be given two operations called **performLaunching(String start)** and **performBlasting(String end)**. Assume this class also works according to the **Singleton** design pattern. [12 marks]