



Sri Lanka Institute of Information Technology

## B. Sc. Honours Degree in Information Technology

Final Examination (Online)

Year 2, Semester I (2020)

IE2021 – Object Oriented Programming

Duration: 3 Hours

### Instructions to Candidates:

- ❖ This paper contains **Three** questions. **Answer All** Questions.
- ❖ Marks for each question are given in the paper.
- ❖ Total Marks: 100.
- ❖ Create a separate Project for each question. The name of the project is provided. Save each Java program using the class name given.
- ❖ This paper contains **05** pages with the Cover Page.

### Instructions to Candidates when submitting:

- ❖ Save all your work.
- ❖ Create a folder from your student ID.
- ❖ Inside that, create 3 separate folders from the project name provided.
- ❖ Copy each project answer source codes( Only the .java files) in to respective folders.(There should be 3 folders name as **Question01, Question02, Question03** inside your ID folder, and in each folder should contain the answer.( .JAVA files ONLY).
- ❖ Zip the Student ID folder (Zip folder also should be the Student ID number).
- ❖ Upload into the correct link.

## Question 1

(35 marks)

This question is based on the **Object-Oriented Programming (OOP) concepts**. You are going to implement a code for a “MenInblack” video game. There is an Alien class to represent a monster and an AlienPack class that represents a pack including different types of aliens. MenInBlack class has different types of agents who is going to kill the aliens and obtain the scores.

a) You can refer the output given in **Main** class and adjust your code accordingly.

```
public class Main {  
  
    public static void main(String args[]) {  
  
        //creating a alien pack with 5 different aliens  
        AlienPack pack1 = new AlienPack(5);  
        pack1.addAlien(new MarshmalloAlien(), 0);  
        pack1.addAlien(new OgreAlien(), 1);  
        pack1.addAlien(new OgreAlien(), 2);  
        pack1.addAlien(new SnakeAlien(), 3);  
        pack1.addAlien(new MarshmalloAlien(), 4);  
  
        MenInBlack AgentK = new MenInBlack(pack1);  
        AgentK.kill();  
        System.out.println("Your score is " + AgentK.getScore());  
    }  
}
```

out - Final2020 (run) ✖ Test Results

run:  
Your score is 55  
BUILD SUCCESSFUL (total time: 0 seconds)

- i). Implement the **Alien** interface and declare **getScore()** method. (01 marks)
- ii). Create three classes called **MarshmalloAlien**, **OgreAlien** and **SnakeAlien** and implement the **Alien** interface in each class. MarshmalloAlien has 15, OgreAlien has 10 and SnakeAlien has 5 as their scores. (3 X 1 = 03 marks)
- iii). Similarly create a class called **AlienPack** and implement the property of **aliens array (Alien[])**. (03 marks)

- iv). Implement the **constructor** in the AlienPack class. Distinguish the number of aliens.  
(03 marks)
  - v). Implement the method called **addAlien** which accept a specific alien and the index.  
(03 marks)
  - vi). Implement the method called **getAliens()** which returns the alien array.  
(03 marks)
  - vii). Implement the **MenInBlack** class and implement the property of **score** and **alienpack(AlienPack)**.  
(03 marks)
  - viii). Implement the **constructor** in the MenInBlack class, which accept and initialize the alienpack. Make the starting agent score as zero.  
(02 marks)
  - ix). Implement the getters and setters for score and alienpack.  
(03 marks)
  - x). Implement the **kill()** method which kills the aliens in the alienpack and calculate the total score depending on each alien.  
(03 marks)
- b) “MenInblack” video game has another level where while the agent is killing the aliens, alien will do a damage for the agent and he will reduce the score accordingly.
- i). Implement the **MenInBlackLevel2** class which is a child of the **MenInBlack** class.  
(02 marks)
  - ii). Implement the **constructor** in the **MenInBlackLevel2** class, which accept and initialize the alienpack.  
(02 marks)
  - iii). Override the kill method where it calculate the total score depending on each alien. Additionally, generate a random number for each alien. Each time the random number is an ODD, the total score will be reduce by 2.  
*Hint: Math.random() \*10 will generate a random decimal number between 1 and 10.*  
(04 marks)

Save the project as **Question01**

## Question 2

(30 marks)

---

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

- i. You should implement a generic class, call MyMathClass , with a type and value parameter T, V where V is a numeric object type (e.g., Integer, Double, or any class that extends java.lang.Number )  
(08 marks)
- ii. Implement a method named Average that takes a HashMapt of type T, V and calculate the average of the HashMap values and display.

*Hint: use doubleValue () method in the Number class to retrieve the value of each number as a double.*

(07 marks)

- iii. Implement another method call ConvertTo which convert and store HashMap values to an ArrayList. Method should takes a HashMapt of type T, V and return the ArrayList.  
(08 marks)
- iv. Implement a class call Mymain which having the main method and test Average and ConvertTo methods with suitable data. Your program should generate a compile-time error if your Average is invoked on a HashMap that is defined for nonnumeric elements as value parameter (e.g., <Strings, String>).

(07 marks)

Save the project as **Question02**

### Question 3

(35 marks)

This question is based on the **Exception Handling**.

a) A program is required to process students marks in an examination. Implement the following classes that makes use of exception handling.

i) Implement a **user defined exception class** called EmailException.

- 1) Have a property called email.
- 2) Implement a **constructor** to get the email property as a parameter and initialize it
- 3) Implement a **getter** for the email property.

(05 marks)

ii) Implement a **class** called User.

- 1) Have the following **properties** as id, name, emails[] array.

(05 marks)

- 2) Implement a **constructor** to get values for the properties id, and name as parameters and initialize them.

(05 marks)

- 3) Implement a **method** called ValidateEmails which return a Boolean and accept the index as the argument.

Note : A valid email should ends with @yahoo.com | @gmail.com | @hotmail.com.  
Any email other than mention suffix should throw an EmailException.  
index is an integer which is greater than zero.

Ignore any errors that can occur through keyboard Input.

(10 marks)

- 4) Implement a method called void input() which allows you to enter all the emails of a user.

i Input values for the emails and handle EmailException.

(05 marks)

iii) Implement a class called MainApp which has a main() function.

- 1) Create a suser object.
- 2) Call the input() method

(05 marks)

Save the project as **Question03**

---

**End of The Examination Paper**

---