Universiti Tun Hussein Onn Malaysia

UNIVERSITI TUN HUSSEIN ONN MALAYSIA

FINAL EXAMINATION
SEMESTER II
SESSION 2023/2024

COURSE NAME	:	VISUAL PROGRAMMING
COURSE CODE	:	BIE 20404
PROGRAMME CODE	:	BIP
EXAMINATION DATE	:	JULY 2024
DURATION	:	3 HOURS
INSTRUCTIONS	:	1. ANSWER ALL QUESTIONS 2. THIS FINAL EXAMINATION IS CONDUCTED VIA ☐ Open book ☒ Closed book 3. STUDENTS ARE **PROHIBITED** TO CONSULT THEIR OWN MATERIAL OR ANY EXTERNAL RESOURCES DURING THE EXAMINATION CONDUCTED VIA CLOSED BOOK

THIS QUESTION PAPER CONSISTS OF **FOUR (4)** PAGES

Q1 Indicate whether each of the following statements is **TRUE** or **FALSE**.

(a) All methods in an abstract class must be declared as abstract methods.

(2 marks)

(b) A `finally` block can be preceded by a `try` block or a `catch` block.

(2 marks)

(c) Every statement must be enclosed in a `try..catch..finally` block.

(2 marks)

(d) To implement multi-threading, a class must implement Runnable and extends Thread class.

(2 marks)

(e) To obtain a thread safe result, an operation must be serialized.

(2 marks)

Q2 Answer **Q2(a)** to **Q2(c)** based on **Figure Q2.1** and **Figure Q2.2**.

```
import javax.swing.JOptionPane;

public class Circle {

	public static void main(String[] args) {
		double radius;

		radius = Double.parseDouble(
			JOptionPane.showInputDialog("Please enter the radius:"));

		double area = radius * radius * 3.14159;

		System.out.println("Circle area is:" + area);
	}
}
```

Figure Q2.1

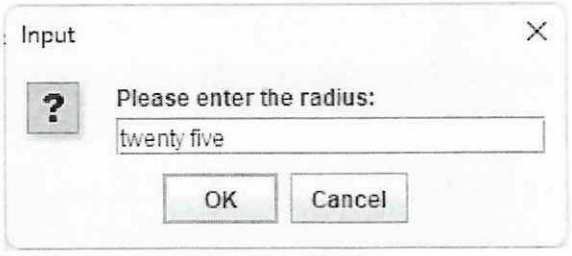

Figure Q2.2

(a) Based on the code in **Figure Q2.1**, if the user enters the input as in **Figure Q2.2** and click the OK button:

(i) What is the output of the program?

(3 marks)

(ii) Justify your answer.

(3 marks)

(b) Apply exception handling for the segment code in **Figure Q2.1** by using a `try..catch` block and a `showMessageDialog()` method.

(8 marks)

(c) Compare between `try..catch` block and `try..finally` block.

(6 marks)

Q3 **Figure Q3.1** shows a single-threaded Java program that simulates a simple task.

```
public class SingleThreadedTask {
    public static void main(String[] args) {
        long startTime = System.currentTimeMillis();

        // Simulate a time-consuming task
        for (int i = 0; i < 10000; i++) {
            System.out.println("Executing step " + i);
            // Simulating a time delay
            try {
                Thread.sleep(1000);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }

        long endTime = System.currentTimeMillis();
        long totalTime = endTime - startTime;
        System.out.println("Total execution time: " + totalTime + "ms");
    }
}
```

Figure Q3.1

(a) Suggests a Java concurrency mechanism to achieve multi-threaded implementation of the task given in **Figure Q3.1**.

(2 marks)

(b) Based on **Q3(a)**, justify your answer.

(4 marks)

(c) Modify the provided program in **Figure Q3.1** to create a multi-threaded version of the task. Each thread should execute a portion of the original task concurrently. Ensure that the multi-threaded implementation maintains the correctness of the original task.

(14 marks)

Q4 Consider a simple database schema for table `Employee`, shown in **Figure Q4.1**.

Employee	
ID (INT)	Name (VARCHAR)

Figure Q4.1

Create a full Java program that connects to the database using JDBC. Implement methods to perform the following operations using Statement interface.

- Insert a new employee into the database.
- Update the name of an employee based on their ID.
- Retrieve and display the details of all employees from the database.

Handle any necessary exception handling within your program. Ensure proper resource management by closing database connections, statements, and result sets.

(20 marks)

Q5 The JDBC `Statement`, `CallableStatement` and `PreparedStatement` interfaces define the methods and properties that enable developers to send SQL commands and receive data from the database. Suggest situations in which each interface should be employed.

(10 marks)

- END OF QUESTIONS -