Name : Arjun P Dinesh
Reg No : 2348506

Question 1:

```
package JavaLabTest2;

import java.util.*;

// this program contain 3 synchronized threads
// first thread prints "Welcome to Java Programming Practical Test2"
// second thread is dedicated to merging overlapping intervals,
// ensuring the output contains only mutually exclusive intevals
// third class determines whether the given strings are anagrams
class QuestionOne {
public static void main(String[] args) {
A a = new A();
B b = new B();
C c = new C();
a.start();
b.start();
c.start();
}
class A extends Thread {
public void run() {
System.out.println("Welcome to Java Programming Practical Test2");
}
}
class B extends Thread {
public void run() {
int[] intervals = {1, 3, 2, 4, 6, 8, 9, 10}
int[] mergedIntervals = MergeOverlappingIntervals.mergeOverlappingIntervals(intervals);
System.out.println("Merged Intervals: " + Arrays.toString(mergedIntervals));
}
}
class C extends Thread {
public void run() {
Scanner myObj = new Scanner(System.in);
System.out.println("Enter String one");
String str1 = myObj.nextLine();
System.out.println("Enter String two");
String str2 = myObj.nextLine();
boolean result = Anagram.areAnagrams(str1, str2);
System.out.println("The strings are " + (result ? "anagrams" : "not anagrams"));
}
}
//MergeOverlappingIntervals - merging overlapping intervals
public class MergeOverlappingIntervals {
public static int[] mergeOverlappingIntervals(int[] intervals) {
```

```
}
}
public class Anagram {

}
}
```

Question 2:

```
package JavaLabTest2;

import java.util.*;
// Question two is to create TreeMap to manage a list of students and their grades at Christ University
// TreeMap should store the student names in alphabetical order corresponding with ttheir grades
// it should automatically maintain names in sorted order
//

public class QuestionTwo {
String name;
int grade;
public void Student(String name, int grade){
this.name = name;
this.grade = grade;
}
class Grade {
int grade;
public Grade(int grade) {
this.grade = grade;
}
}

public class StudentGradeManager {
private TreeMap<Student, Grade> studentGrades;
public StudentGradeManager() {
studentGrades = new TreeMap<>();
}
public void addStudent(Student student) {
studentGrades.put(student, null);
```

```
}
public void viewStudent(Student student) {
System.out.println(studentGrades.get(student));
}
public void updateGrade(Student student, Grade newGrade) {
public static void main(String[] args) {
StudentGradeManager manager = new StudentGradeManager();
Scanner scanner = new Scanner(System.in);
while (true) {
System.out.println("Menu:");
System.out.println("1. Add Student");
System.out.println("2. View Student");
System.out.println("3. Update Grade");
System.out.println("4. Remove Student");
System.out.println("5. Search Student");
System.out.println("6. Exit");
int choice = scanner.nextInt();
scanner.nextLine();
switch (choice) {
case 1:
System.out.print("Enter student name: ");
String name = scanner.nextLine();
System.out.print("Enter student ID: ");
int id = scanner.nextInt();
scanner.nextLine(); // consume newline left-over
manager.addStudent(new Student());
break;
case 2:
System.out.print("Enter student name: ");
name = scanner.nextLine();
Student student = manager.searchStudent(name);
if (student != null) {
manager.viewStudent(student);
} else {
System.out.println("Student not found");
}
break;
}
}
}

private Student searchStudent(String name) {
return null;
}
}}
```