

## Question 1

```
public class Main {

    public static boolean dividesSelf(int val) {
        int i = val;
        while( i!=0 ) {
            if( (i%10 == 0) || (val % (i%10)) != 0) {
                return false;
            }
            i/=10;
        }
        return true;
    }

    public static void main(String[] args) {
        System.out.println(dividesSelf(128)); // True
        System.out.println(dividesSelf(12)); // True
        System.out.println(dividesSelf(120)); // False
    }
}
```

## Question 2

Module.java

```
public class Module {

    private String code;
    private String title;
    private double result; // in percentage

    public void setResult(double result) {
        if(result>100) {
            this.result = 100;
        }
        else if(result<0) {
            this.result = 0;
        }
        else this.result = result;
    }

    public double getResult() {
        return result;
    }

    public Module(String code, String title, double result) {
        this.code = code;
        this.title = title;
        this.setResult(result);
    }

    public void display() {
        System.out.println("code: " + code);
        System.out.println("title: " + title);
        System.out.println("result: " + result + "%");
    }
}
```

## Student.java

```
public class Student {

    private String name;
    private int id_number;
    private String programme;
    private Module[] modules;

    public Student(String name, int id_number, String programme,
                   Module[] modules) {
        this.name = name;
        this.id_number = id_number;
        this.programme = programme;
        this.modules = modules;
    }

    public void display() {
        System.out.println("name: " + name);
        System.out.println("id_number: " + id_number);
        System.out.println("programme: " + programme);
        System.out.println(modules.length + " modules: ");
        System.out.println("-----");
        for( int i=0; i<modules.length; i++ ) {
            modules[i].display();
            System.out.println("-----");
        }
    }
}
```

## Main.java

```
public class Main {

    public static void main(String[] args) {
        Module[] modules = {new Module("EE219", "OOP", 98.7),
                            new Module("EE42", "Chuck Norris", 101)};

        Student luke = new Student("Skywalker", 1234, "eeng", modules);
        luke.display();
    }
}
```

## Question 3

Added to Student.java

```
    public double getAverage() {
        double average=0;
        for(int i=0; i<modules.length; i++) {
            average += modules[i].getResult() / modules.length;
        }
        return average;
    }
}
```

```

public String getGrade() {
    double average = getAverage();
    if(average >= 70.0) {
        return "H1";
    }
    else if(average >= 60.0) {
        return "H2.1";
    }
    else if(average >= 50.0) {
        return "H2.2";
    }
    else if(average >= 40.0) {
        return "H3";
    }
    else {
        return "failed";
    }
}

```

Main.java

```

public class Main {

    public static void main(String[] args) {
        Module[] modules = {new Module("EE219", "OOP", 50.0),
                             new Module("EE42", "Chuck Norris", 100.0)};

        Student luke = new Student("Skywalker", 1234, "eeng", modules);
        System.out.println("average : " + luke.getAverage() + "%");
        System.out.println("grade : " + luke.getGrade());
    }
}

```

## Question 4

Added in Module.java

```

public class Module implements Comparable<Module> {

    ...

    @Override
    public int compareTo(Module o) {
        if(this.getResult() - o.getResult() < 0) {
            return -1;
        }
        else if(this.getResult() - o.getResult() > 0) {
            return 1;
        }
        else return 0;
    }
}

```

### Added in Student.java

```
import java.util.Arrays;

public class Student {

    ...

    public Student(String name, int id_number, String programme,
                   Module[] modules) {
        this.name = name;
        this.id_number = id_number;
        this.programme = programme;
        this.modules = modules;
        Arrays.sort(modules); // sorting modules for highest/lowest/median
    }

    ...

    public Module getLowest() {
        return modules[0];
    }

    public Module getHighest() {
        return modules[modules.length-1];
    }

    public Module getMedian() {
        return modules[Math.round(modules.length/2)];
    }
}
```

### Main.java

```
public class Main {

    public static void main(String[] args) {

        Module[] modules = {new Module("EE219", "OOP", 50.0),
                            new Module("EE42", "Chuck Norris", 100.0),
                            new Module("EE123", "low module", 25.0)};

        Student luke = new Student("Skywalker", 1234, "eeng", modules);

        System.out.print("Lowest module : ");
        luke.getLowest().display();
        System.out.print("Median module : ");
        luke.getMedian().display();
        System.out.print("Highest module : ");
        luke.getHighest().display();
    }
}
```

## Question 5

### Added in Student.java

```
import java.util.Arrays;

public class Student implements Comparable<Student> {

    ...

    public String getName() {
        return name;
    }

    @Override
    public int compareTo(Student o) {
        if(this.getAverage() - o.getAverage() < 0) {
            return -1;
        }
        else if(this.getAverage() - o.getAverage() > 0) {
            return 1;
        }
        else return 0;
    }
}
```

### Programme.java

```
import java.util.Arrays;

public class Programme {

    private String title = "unknow";
    private Student[] students;

    public Programme(String title, Student[] students) {
        this.title = title;
        this.students = students;
        Arrays.sort(this.students);
    }

    public Programme(Student[] students) {
        this.students = students;
        Arrays.sort(this.students);
    }

    public void display() {
        System.out.println("Title: " + title);
        for(int i=0; i<students.length; i++) {
            students[i].display();
        }
    }

    public Student getWorst() {
        return students[0];
    }

    public Student getBest() {
        return students[students.length-1];
    }
}
```

## Main.java

```
public class Main {

    public static void main(String[] args) {

        Module[] modules1 = {new Module("EE219", "OOP", 50.0),
                               new Module("EE42", "Chuck Norris", 100.0),
                               new Module("EE123", "low module", 25.0)};

        Module[] modules2 = {new Module("EE219", "OOP", 10.0),
                               new Module("EE42", "Chuck Norris", 70.0),
                               new Module("EE123", "low module", 35.0)};

        Student[] students = {new Student("Skywalker", 1234, "eeng", modules1),
                               new Student("Yoda", 5678, "eeng", modules2)};

        Programme prog = new Programme(students);
        prog.display();

        System.out.println("Best student: " + prog.getBest().getName());
        System.out.println("Worst student: " + prog.getWorst().getName());
    }
}
```

## Question 6

```
public class Main {

    public static int[] lotteryNumbers() {

        int rand = 0;
        Boolean is_duplicate = false;
        int[] output = { 0, 0, 0, 0, 0, 0, 0 };

        for (int i = 0; i < 7; i++) {
            do {
                rand = (i < 5) ? (int) (50 * Math.random() + 1)
                               : (int) (11 * Math.random() + 1);
                is_duplicate = false;
                for (int j = 0; j < i; j++) {
                    if (rand == output[j]) {
                        is_duplicate = true;
                    }
                }
            } while (is_duplicate);
            output[i] = rand;
        }
        return output;
    }

    public static void main(String[] args) {
        int[] numbers = lotteryNumbers();
        for (int i = 0; i < 7; i++)
            System.out.println(numbers[i]);
    }
}
```

## Question 7

```
import java.awt.*;
import java.awt.event.*;

public class LotteryApplication extends Frame implements ActionListener {
    Button lotteryButton;
    TextField[] lotteryfileds = new TextField[7];

    public static int[] lotteryNumbers() {

        int rand = 0;
        Boolean is_duplicate = false;
        int[] output = { 0, 0, 0, 0, 0, 0, 0 };

        for (int i = 0; i < 7; i++) {
            do {
                rand = (i < 5) ? (int) (50 * Math.random() + 1)
                    : (int) (11 * Math.random() + 1);
                is_duplicate = false;
                for (int j = 0; j < i; j++) {
                    if (rand == output[j]) {
                        is_duplicate = true;
                    }
                }
            } while (is_duplicate);
            output[i] = rand;
        }
        return output;
    }

    public LotteryApplication() {

        super("Lottery Generator");
        this.setLayout(new FlowLayout());
        lotteryButton = new Button("Lottery");
        this.add(lotteryButton);
        lotteryButton.addActionListener(this);

        for (int i = 0; i < 7; i++) {
            lotteryfileds[i] = new TextField(2);
            lotteryfileds[i].setEditable(false);
            this.add(lotteryfileds[i]);
        }
        this.setSize(400, 100);
        this.setVisible(true);
    }

    public static void main(String[] args) {
        LotteryApplication myApp = new LotteryApplication();
    }

    public void actionPerformed(ActionEvent arg0) {
        int[] result = lotteryNumbers();
        for (int i = 0; i < result.length; i++) {
            lotteryfileds[i].setText(new Integer(result[i]).toString());
        }
    }
}
```

