

MATRIC NUMBER: 20172451

QUESTION 1.

```
package question;
import java.text.DecimalFormat;
import java.util.Scanner;

public class Question {

    /**
     * @param args the command line arguments
     */
    public static void inputData(float a[], int N) {
        Scanner sc = new Scanner(System.in);
        int i;
        for(i=0;i<N;i++) {
            System.out.println("Input the value of "+i);
            a[i] = sc.nextFloat();
        }
    }

    public static void maximum(float a[], int N) {
        int i;
        float max = a[0];

        for(i=0;i<N;i++) {
            if(a[i]> max) {
                max = a[i];
            }
        }
        System.out.println("Maximum is"+max);
    }

    public static void minimum(float a[], int N) {
        int i;
        float min = a[0];

        for(i=0;i<N;i++) {
            if(a[i]< min) {
                min = a[i];
            }
        }
        System.out.println("Minimum is"+min);
    }
}
```

```

    public static void minimum(float a[], int N) {
        int i;
        float min = a[0];

        for(i=0;i<N;i++) {
            if(a[i]< min) {
                min = a[i];
            }
        }
        System.out.println("Minimum is"+min);

    }

    public static void variation(float a[], int N) {
        DecimalFormat df = new DecimalFormat("##.##");
        float var = 0; double sd;float sum = 0;
        for(var i = 0;i<N;i++) {
            sum = sum + a[i];
        }
        float avg = sum/N;
        for(var k=0;k<N;k++) {
            var = var + ((a[k]- avg)*(a[k]- avg))/N;
        }

        System.out.println("The variance is "+ var);
        System.out.println("variance after format is"+ df.format(var));
        sd = Math.sqrt(var);
        System.out.println("The standard deviation is "+ sd);
        System.out.println("The standard deviation after format is "+
df.format(sd));
    }

    public static void mean(float a[], int N) {
        DecimalFormat df = new DecimalFormat("##.##");
        float mean; float fx; float x;
        x = 0; int i;
        for(i=0;i<N; i++) {
            x = x + a[i];
        }
        fx = x * N;
        mean = fx / N;
        // fmean = df.format(mean);
        System.out.println("The mean is "+ mean);

    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
    }

```

```

        System.out.print("Input the length of the array");
        int N = sc.nextInt();
        if(N < 10) {
            System.out.print("The length of the array must be at least
10");
            N = sc.nextInt();
        }
        float a[] = new float[N];
        inputData(a,N);
        maximum(a,N);
        minimum(a,N);
        variation(a,N);
        mean(a,N);
        System.out.println("Working fine");
    }

}

```

QUESTION 5

```

/*
 * Click
nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to
change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/GuiForms/JPanel.java to
edit this template
 */
package question5;

/**
 *
 * @author DYNAMIX
 */
public class NewJFrame extends javax.swing.JPanel {

    /**
     * Creates new form NewJFrame
     */
    public NewFrame() {
        initComponents();
    }

    /**
     * This method is called from within the constructor to initialize the
form.
     * WARNING: Do NOT modify this code. The content of this method is
always

```

```

    * regenerated by the Form Editor.
    */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        single = new javax.swing.JRadioButton();
        married = new javax.swing.JRadioButton();
        divorce = new javax.swing.JRadioButton();
        seperation = new javax.swing.JRadioButton();
        remarried = new javax.swing.JRadioButton();
        cohabitation = new javax.swing.JRadioButton();
        windowhood = new javax.swing.JRadioButton();
        check = new javax.swing.JButton();
        result = new javax.swing.JTextField();

        single.setText("single");

        married.setText("married");

        divorce.setText("divorce");

        seperation.setText("seperation");

        remarried.setText("remarried");

        cohabitation.setText("cohabitation");

        windowhood.setText("windowhood");

        check.setText("check status");
        check.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                checkActionPerformed(evt);
            }
        });

        result.setText("result");

        javax.swing.GroupLayout layout = new
    javax.swing.GroupLayout(this);
        this.setLayout(layout);
        layout.setHorizontalGroup(

    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .add(layout.createParallelGroup()
                .add(single)
                .add(married)
                .add(divorce)
                .add(seperation)
                .add(remarried)
                .add(cohabitation)
                .add(windowhood)
                .add(check)
                .add(result)
            )
            .addContainerGap(10, true)
        )
    );
    
```

```

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

        .addComponent(windowhood)
        .addComponent(cohabitation)
        .addComponent(remarried)
        .addComponent(seperation)
        .addComponent(divorce)
        .addComponent(married)
        .addComponent(single))
.addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
.addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
        .addContainerGap(177, Short.MAX_VALUE)
        .addComponent(check)
        .addGap(18, 18, 18)
        .addComponent(result,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(72, 72, 72))
);
layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
                .addGap(26, 26, 26)
                .addComponent(single)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addComponent(married)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addComponent(divorce)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addComponent(seperation)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addComponent(remarried)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(cohabitation)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(windowhood)

```

```

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

        .addComponent(check)
        .addComponent(result,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap(51, Short.MAX_VALUE))

    );
} // </editor-fold>

private void checkActionPerformed(java.awt.event.ActionEvent evt) {
    ButtonGroup bg = new ButtonGroup();
    bg.add(single);
    bg.add(married);
    bg.add(divorce);
    bg.add(seperation);
    bg.add(remarried);
    bg.add(cohabitation);
    bg.add(windowhood);
    if(single.isSelected() == true) {
        result.setText(single.getText());
    }
    if(married.isSelected() == true) {
        result.setText(married.getText());
    }
    if(divorce.isSelected() == true) {
        result.setText(divorce.getText());
    }
    if(seperation.isSelected() == true) {
        result.setText(seperation.getText());
    }
    if(remarried.isSelected() == true) {
        result.setText(remarried.getText());
    }
    if(cohabitation.isSelected() == true) {
        result.setText(cohabitation.getText());
    }
    if(windowhood.isSelected() == true) {
        result.setText(windowhood.getText());
    }
}

```

```

        // Variables declaration - do not modify
        private javax.swing.JButton check;
        private javax.swing.JRadioButton cohabitation;
        private javax.swing.JRadioButton divorce;
        private javax.swing.JRadioButton married;
        private javax.swing.JRadioButton remarried;
        private javax.swing.JTextField result;
        private javax.swing.JRadioButton seperation;
        private javax.swing.JRadioButton single;
        private javax.swing.JRadioButton windowhood;
        // End of variables declaration
    }

```

QUESTION 2

```

/*
 * Click
 nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to
 change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to
 edit this template
 */
package question2;
import java.util.Scanner;
/**
 *
 * @author DYNAMIX
 */
public class Question2 {
    public static int hcf(int a, int b) {
        int n1 =a; int n2=b; int temp;
        while(n2 > 0) {
            temp = n2;
            n2 = n1 % n2;
            n1 = temp;
        }
        return n1;
    }
    public static float lcm(int a, int b) {
        int n1 =a; int n2=b; int temp;
        while(n2 > 0) {
            temp = n2;
            n2 = n1 % n2;
            n1 = temp;
        }
        //      int good = Integer.parseInt(n1/2);

```

```

        return n1/2;
    }
    public static int difference(int a, int b) {
        int n1 = a; int n2 = b;
        int lcmd = lcm(n1,n2);
        int hcfd = hcf(n1,n2);
        int difference = lcmd -hcfd;
        return difference;
    }
    public static int greatest(int a, int b) {
        int n1 = a; int n2 = b;
        if(n1 > n2) {
            return n1;
        }
        else {
            return n2;
        }
    }
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Input first number");
        int a = sc.nextInt();
        System.out.print("Input second number");
        int b = sc.nextInt();
        System.out.print("Input third number");
        int c = sc.nextInt();

        int firsthcf = hcf(a,b);
        int finalhcf = hcf(firsthcf,c);
        System.out.println("The HCF is" + finalhcf);
        float firstlcm = lcm(a,b);
        float finallcm = lcm(firstlcm,c);
        System.out.println("The LCM is" + finallcm);
        int firstgreatest = greatest(a,b);
        int finalgreatest = greatest(firstgreatest,c);
        System.out.println("The greatest is" + finalgreatest);
        int cdifferencel = difference(a,b);
        int finaldifference = difference(cdifferencel,c);
        System.out.println("The difference between LCM and HCF is" +
finaldifference);

    }

}

```