



Faculty of Science

Course: CSCI 2020u – Software System Development & Integration

Component: Assignment

Weight: 10%

Deadline: March 6, 2019 (due by 11:59pm)

Collaboration Policy

You are permitted to work on this assignment in a team, and submit the results as a team. For this sort of assignment, with an open-ended component, the collaboration between multiple team members can be beneficial. Between groups, however, please limit the discussion to the level of general strategy (not code). Groups of size 2 are recommended. Larger groups will be considered with the proviso that the marker will mark your assignment with higher expectations. In any case, be sure that all members of the team fully understand all code, otherwise they will miss intended learning objectives, which may be a considerable disadvantage at exam time.

How to Submit

You will maintain a **git repository** for this assignment, which is a public repository. To submit the assignment, create a single file 'README.txt' that contains instructions on how to download, compile, and run your codes for each question. A .zip, .7z, or .rar file will not be acceptable. **Also submit this word file (once you complete) into related drop box on Blackboard before deadline.**

Note: *Comments are mandatory. Failure to properly document your program will result in a deduction on the marks you receive for this (and any other) assignment.*

Remember:

You need to complete this file and submit it in related **drop box on Blackboard**, in addition to uploading your codes in your **git repository**, before deadline.

Question 1: Displaying Three Cards

Problem Description:

Display a frame that contains three labels. Each label displays a card, as shown in the figure below. The card image files are named 1.png, 2.png, ..., 54.png and stored in the image/card directory. All three cards are distinct and selected randomly.

The image icons can be found in the attached card folder.



Your Task:

1. Create three ImageView and set their icons using the images.
2. Display three images from 54 image cards randomly.

Your Code:

Copy-paste your code here:

```
//Gajan Sivanesan
//100425203
//Assignment 1 Question 1
//Due Date: March/6/2019

import java.util.ArrayList;
import javafx.application.Application;
import javafx.scene.Scene;
import javafx.scene.image.Image;
import javafx.scene.image.ImageView;
import javafx.scene.layout.HBox;
import javafx.stage.Stage;

public class Question1 extends Application {

    public static void main(String[] args) {
        Application.launch(args);
    }
}
```

```

@Override
public void start(Stage primaryStage) {

    // initialize a card dynamic array
    ArrayList<Integer> cards = new ArrayList<>();

    // fill the array with the string values of numbers since the
    cards have numbers in their name.
    for (int i = 0; i < 52 ; i++) {
        cards.add(i+1);
    }
    // function that randomizes the array
    java.util.Collections.shuffle(cards);

    ImageView viewCard1 = new ImageView(new Image(
"Cards/"+cards.get(0) + ".png"));
    ImageView viewCard2 = new ImageView(new
Image("Cards/"+cards.get(1) + ".png"));
    ImageView viewCard3 = new ImageView(new Image
("Cards/"+cards.get(2) + ".png"));

    HBox display = new HBox();

    display.getChildren().add(viewCard1);
    display.getChildren().add(viewCard2);
    display.getChildren().add(viewCard3);

    Scene scene = new Scene(display, 230, 100);

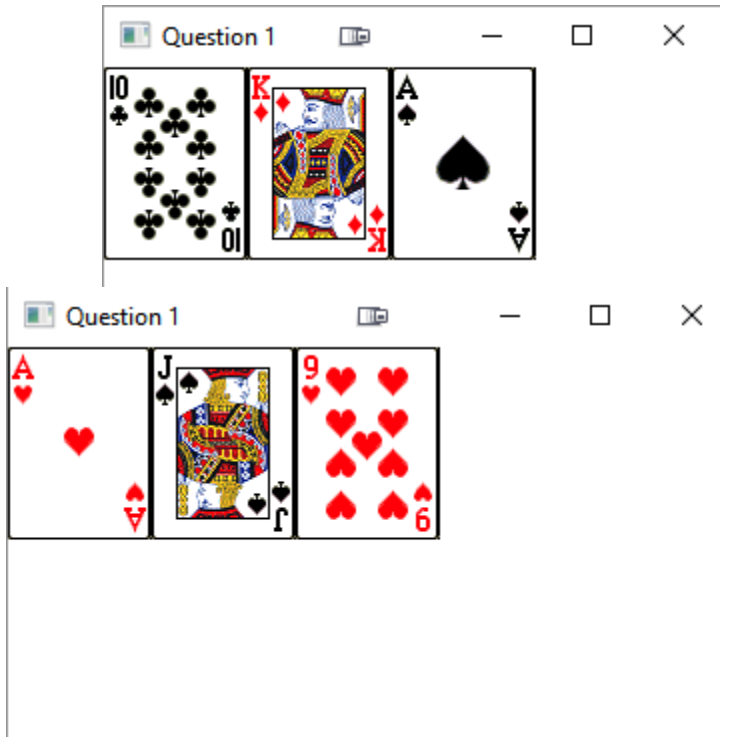
    primaryStage.setTitle("Question 1");
    primaryStage.setScene(scene);
    primaryStage.show();

}
}

```

Screen shots:

Include two screen shots here:

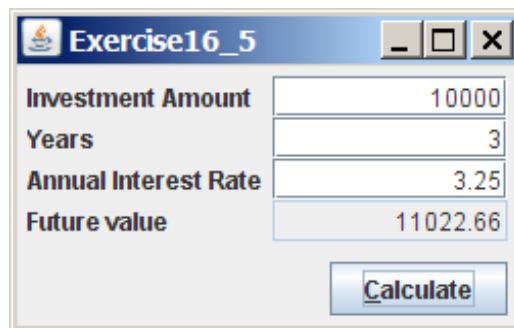


Question 2: Investment-Value calculator

Problem Description:

Write a program that calculates the future value of an investment at a given interest rate for a specified number of years. The formula for the calculation is as follows:

$$\text{futureValue} = \text{investmentAmount} * (1 + \text{monthlyInterestRate})^{\text{years} * 12}$$



Investment Amount	10000
Years	3
Annual Interest Rate	3.25
Future value	11022.66

Calculate

Your Task:

Use text fields for interest rate, investment amount, and years. Display the future amount in a text field when the user clicks the Calculate button, as shown in the figure.

Your Code:

Copy-paste your code here.

```
// Gajan Sivanesan
// 100425203
// Assignment 1 Question 2
// Due Date: March/06/2019

import javafx.application.Application;
import javafx.event.ActionEvent;
import javafx.event.EventHandler;
import javafx.geometry.Insets;
import javafx.scene.Scene;
import javafx.scene.control.*;
import javafx.scene.layout.GridPane;
import javafx.stage.Stage;
import java.text.DecimalFormat;

public class Question2 extends Application {

    private TextField _investmentAmount;
```

```

private TextField _years;
private TextField _anInterestRate;
private Button _calculate;
private TextField _futureValue;

private static DecimalFormat df2 = new DecimalFormat("####0.00");

public static void main(String[] args) {
    Application.launch(args);
}

@Override
public void start(Stage primaryStage) throws Exception {
    primaryStage.setTitle("Question 2");

    // Grid Pane
    GridPane gp = new GridPane();
    gp.setPadding(new Insets(10,10,10,10));

    // investment amount field
    Label investmentLabel = new Label("Investment Amount");
    _investmentAmount = new TextField();
    gp.add(investmentLabel, 0,0);
    gp.add(_investmentAmount, 1, 0);

    // years field
    Label yearsLabel = new Label("Years: ");
    _years = new TextField();
    gp.add(yearsLabel, 0, 1);
    gp.add(_years, 1, 1);

    // annual interest rate field
    Label anInterestRateLabel = new Label("Annual Interest Rate: ");
    _anInterestRate = new TextField();
    gp.add(anInterestRateLabel, 0, 2);
    gp.add(_anInterestRate, 1, 2);

    // future value field
    Label futureValueLabel = new Label("Future Value: ");
    _futureValue = new TextField();
    _futureValue.setDisable(true); // makes the user unable to
interact with the textfield
    gp.add(futureValueLabel, 0,3);
    gp.add(_futureValue, 1, 3);

    _calculate = new Button("Calculate");

```

```

        _calculate.setDefaultButton(true); // sets button to execute when
enter is pressed
        gp.add(_calculate, 1, 4);

        // Button handler
        _calculate.setOnAction(new EventHandler<ActionEvent>() {
            @Override
            public void handle(ActionEvent event) {

                double years = Double.parseDouble(_years.getText());
                double investmentAmount =
Double.parseDouble(_investmentAmount.getText());
                double anInterestRate =
Double.parseDouble(_anInterestRate.getText());

                // formula for calculating the future value
                double futureValue = investmentAmount* (Math.pow((1 +
((anInterestRate/100)/12)), (years*12)));

                String finalValue = Double.toString(futureValue);

                _futureValue.appendText(finalValue);

            }
        });
        Scene scene = new Scene(gp, 300, 150);
        primaryStage.setScene(scene);
        primaryStage.show();
    }
}

```

Screen shots:

Include two screen shots here:

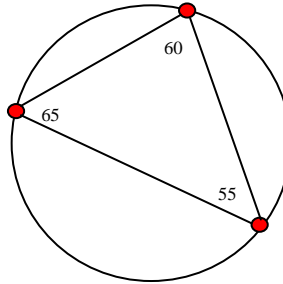
The image displays two screenshots of a software window titled "Question 2". Each window contains a form for calculating the future value of an investment. The top window shows an investment of 10,000 over 3 years at a 3.25% interest rate, resulting in a future value of 11,022.66. The bottom window shows an investment of 50,000 over 25 years at a 10% interest rate, resulting in a future value of 350,360.28. Both windows have a "Calculate" button.

Investment Amount	Years	Annual Interest Rate	Future Value
10000	3	3.25	11022.661606473503
50000	25	10	350360.2847.2511750533

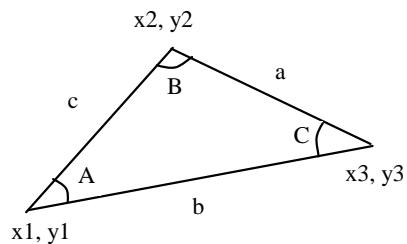
Question 3: Dragging Points on a Circle

Problem Description:

Draw a circle with three random points on the circle. Connect the points to form a triangle. Display the angles in the triangle. Use the mouse to drag a point along the perimeter of the circle. As you drag it, the triangle and angles are redisplayed dynamically.



Here is the formula to compute angles:



$$\begin{aligned} A &= \arccos((a * a - b * b - c * c) / (-2 * b * c)) \\ B &= \arccos((b * b - a * a - c * c) / (-2 * a * c)) \\ C &= \arccos((c * c - b * b - a * a) / (-2 * a * b)) \end{aligned}$$

Your Code:

Copy-paste your code here:

```
// Gajan Sivanesan
// 100425203
// Assignment 1 Question 3
// Due Date: March/06/2019

import javafx.application.Application;
import javafx.stage.Stage;
import javafx.scene.Scene;
import javafx.scene.layout.Pane;
import javafx.scene.shape.Circle;
import javafx.scene.shape.Line;
import javafx.scene.text.Text;
```

```

import javafx.scene.paint.Color;
import javafx.scene.input.MouseButton;

public class Question3 extends Application {

    // calculate the length of each line
    public static double getDistance(Line line) {
        double length;
        double xStart = line.getStartX();
        double xEnd = line.getEndX();
        double yStart = line.getStartY();
        double yEnd = line.getEndY();

        length = Math.sqrt(Math.pow(xStart - xEnd, 2)) +
Math.sqrt(Math.pow(yStart - yEnd, 2));

        return length;
    }

    // calculate the angles
    public static double getAngle(Line x, Line y, Line z) {
        double angle;
        double a = getDistance(x);
        double b = getDistance(y);
        double c = getDistance(z);

        angle = Math.toDegrees(Math.acos((a * a - b * b - c * c) / (-2 * b
* c)));

        return angle;
    }

    public static void main(String args[]) {
        launch(args);
    }

    @Override
    public void start(Stage primaryStage) {
        Circle[] smallCircles = new Circle[3];
        Line[] edges = new Line[3];
        Text[] labels = new Text[3];
        Pane pane = new Pane();
        int next;

        Circle mainCircle = new Circle(500,500,100);

```

```

mainCircle.setFill(Color.WHITE);
mainCircle.setStroke(Color.BLACK);
pane.getChildren().add(mainCircle);

// generate 3 small circles around the big circle
for(int i = 0; i < 3; i++) {
    Circle vertex = new Circle(10);
    double x = mainCircle.getCenterX() +
100*Math.cos(2*Math.PI*Math.random());
    double y = mainCircle.getCenterY() +
100*Math.sin(2*Math.PI*Math.random());

    vertex.setCenterX(x);
    vertex.setCenterY(y);
    smallCircles[i] = vertex;
    vertex.setOnMouseDragged(e -> {

        if(e.getButton().equals(MouseButton.PRIMARY)) {
            double angle = Math.atan2(e.getY() -
mainCircle.getCenterY(), e.getX() - mainCircle.getCenterX()); //
calculates angle

            Circle c = (Circle)e.getSource();
            c.setCenterX(mainCircle.getCenterX() + 100 *
Math.cos(angle));
            c.setCenterY(mainCircle.getCenterY() + 100 *
Math.sin(angle));

            // find angles of each vertex
            double angle1 = getAngle(edges[0], edges[1],
edges[2]);
            double angle2 = getAngle(edges[1], edges[2],
edges[0]);
            double angle3 = getAngle(edges[2], edges[0],
edges[1]);

            // set values to the circles
            labels[0].setText(Double.toString(angle1));
            labels[1].setText(Double.toString(angle2));
            labels[2].setText(Double.toString(angle3));
        }
    });
}

// connect the small circles
for(int i = 0; i < 3; i++) {
    if(i == 2) {
        next = 0;
    }
    else {

```

```

        next = i + 1;
    }
    Line length = new Line();

length.startXProperty().bind(smallCircles[i].centerXProperty());
length.endXProperty().bind(smallCircles[next].centerXProperty());

length.startYProperty().bind(smallCircles[i].centerYProperty());
length.endYProperty().bind(smallCircles[next].centerYProperty());

    edges[i] = length;
}

// set the angles to the vertex
for(int i = 0; i < 3; i++) {
    Text text = new Text();
    text.xProperty().bind(smallCircles[i].centerXProperty());
text.yProperty().bind(smallCircles[i].centerYProperty().subtract(200));

    labels[i] = text;
}

pane.getChildren().addAll(smallCircles[0], smallCircles[1],
smallCircles[2], edges[0], edges[1], edges[2], labels[0], labels[1],
labels[2]);

    Scene scene = new Scene(pane, 900, 900);
    primaryStage.setTitle("Question 3");
    primaryStage.setScene(scene);
    primaryStage.show();

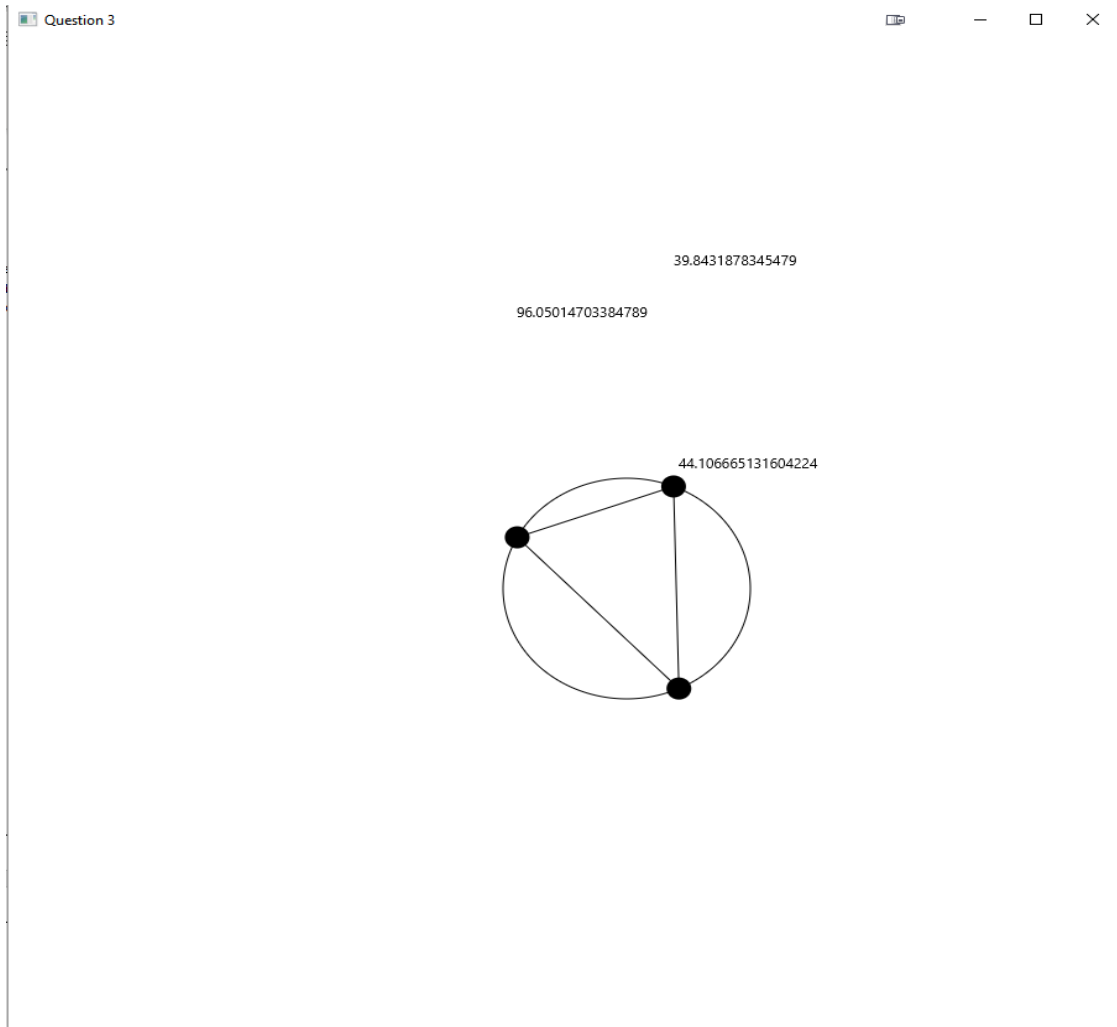
}

}

```

Screen shots:

Include two screen shots here:



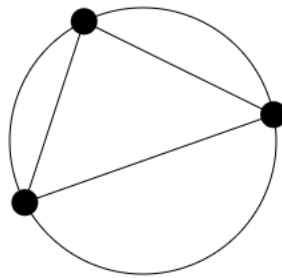
Question 3



55.70873316780216

79.6619341951716

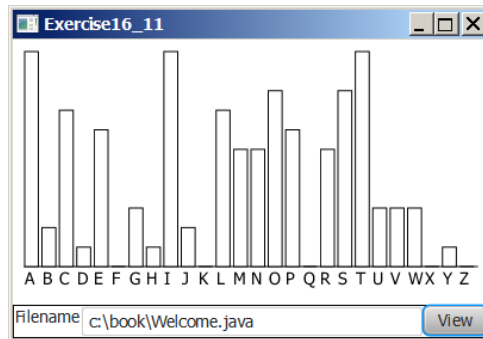
44.62933263702625



Question 4: Histogram

Problem Description:

Develop a program that displays a histogram to show the occurrences of each letter in a text area. The histogram should show the occurrences of each letter in a text file, as shown in the following figure. Assume that the letters are not case sensitive.



Your Task:

- Place a pane that will display the histogram in the center of the frame.
- Place a label and a text field in a panel, and put the panel in the south side of the frame. The text file will be entered from this text field.
- Pressing the Enter key on the text field causes the program to count the occurrences of each letter and display the count in a histogram.

Your Code:

Copy-paste your code here:

```
// Gajan Sivanesan
// 100425203
// Assignment 1 Question 4
// Due Date: March/06/2019

import javafx.application.Application;
import javafx.stage.Stage;
import javafx.scene.Scene;
import javafx.scene.control.Label;
import javafx.scene.control.TextField;
import javafx.scene.control.Button;
import javafx.scene.layout.HBox;
import javafx.scene.layout.BorderPane;
import javafx.geometry.Insets;
import java.io.File;
```

```

import java.net.URI;
import java.util.Scanner;

public class Question4 extends Application {

    private TextField _fileInput;
    private Button _viewBtn;
    private Label _fileInputLabel;

    static int [] finalCount = new int[26];

    // parse using ASCII
    public static void countLetters(String input) {
        for(int i = 0; i < input.length(); i++) {
            char letters = input.charAt(i);

            if(Character.isLetter(letters)) {
                finalCount[(int)letters - 97]++;
            }
        }
    }

    public static void main(String[] args) {
        Application.launch(args);
    }

    @Override
    public void start(Stage primaryStage) throws Exception {

        _fileInputLabel = new Label("File Path:");
        _fileInput = new TextField();
        _viewBtn = new Button("Calculate");
        _viewBtn.setDefaultButton(true); // sets button to execute when
enter is pressed
        Histogram histogram = new Histogram();

        // Button handler
        // I have no Idea how to fix this.
        // I don't quite understand how URI works
        // It works if the file is in the directory and you just type the
filename
        // After further testing it seems to only detect .txt files that
are named Test. It might be due to my operating system
        _viewBtn.setOnAction(e -> {
            finalCount = new int[26];

```



```

        try {
            //String fileURL = _fileInput.getText();
            //URI uri = new URI(fileURL);
            URI uri = new URI(_fileInput.getText());
            File file;

            if (uri.isAbsolute()) {
                file = new File(uri);
            }
            else {
                file = new File(uri.toString());
            }
            // converts text to lower case
            try(Scanner input = new Scanner(file)) {
                StringBuilder builder = new StringBuilder();
                while (input.hasNext()) {
                    builder.append(input.nextLine().toLowerCase()
+ "\n");
                }

                countLetters(builder.toString());
                histogram.setFinalCount(finalCount);
            }
            catch (Exception ex) {
                System.out.println(ex);
            }

        }
        catch (Exception ex) {
            System.out.println(ex);
        }
    });

    HBox box = new HBox(20);
    box.getChildren().addAll(_fileInputLabel, _fileInput, _viewBtn);

    BorderPane pane = new BorderPane();
    pane.setCenter(histogram);
    pane.setBottom(box);
    pane.setMargin(histogram, new Insets(30));

    Scene scene = new Scene(pane);
    primaryStage.setTitle("Question 4");
    primaryStage.setScene(scene);
    primaryStage.show();
}

```

```
}
```

```
// Gajan Sivanesan
// 100425203
// Assignment 1 Question 4
// Due Date: March/06/2019

import javafx.scene.layout.Pane;
import javafx.scene.shape.Rectangle;
import javafx.scene.text.Text;
import javafx.scene.paint.Color;

public class Histogram extends Pane {

    private int[] finalCount;
    private int maxValue;
    private int paneHeight = 600;

    // function to calculate the maximum value
    public int getMaxValue() {
        int j = finalCount[0];
        for (int i = 1; i < finalCount.length; i++) {
            if (finalCount[i] > j ) {
                j = finalCount[i];
            }
        }
        return j;
    }

    public void setFinalCount (int[] finalCount) {
        this.finalCount = finalCount;
        maxValue = getMaxValue();
        generateHistogram();
    }

    public void generateHistogram() {
        int x = 0;
        for (int i = 0; i < finalCount.length; i++) {
            double rectangleHeight = (double)finalCount[i]/maxValue *
paneHeight; // calculate rectangle height
            Rectangle rectangle = new Rectangle(20, rectangleHeight);
            rectangle.setX(x);
            rectangle.setY(paneHeight- rectangleHeight);
            rectangle.setFill(Color.WHITE);
            rectangle.setStroke(Color.BLACK);
        }
    }
}
```

```

        Text text = new Text((char)(i + 65) + " ");
        text.setX(x + 5); // aligns the letters with the bars in the
graph
        text.setY(610); // static y coordinate for all the letters
        getChildren().addAll(rectangle, text);
        x += 20; // increments the x coordinate to space it out
    }
}

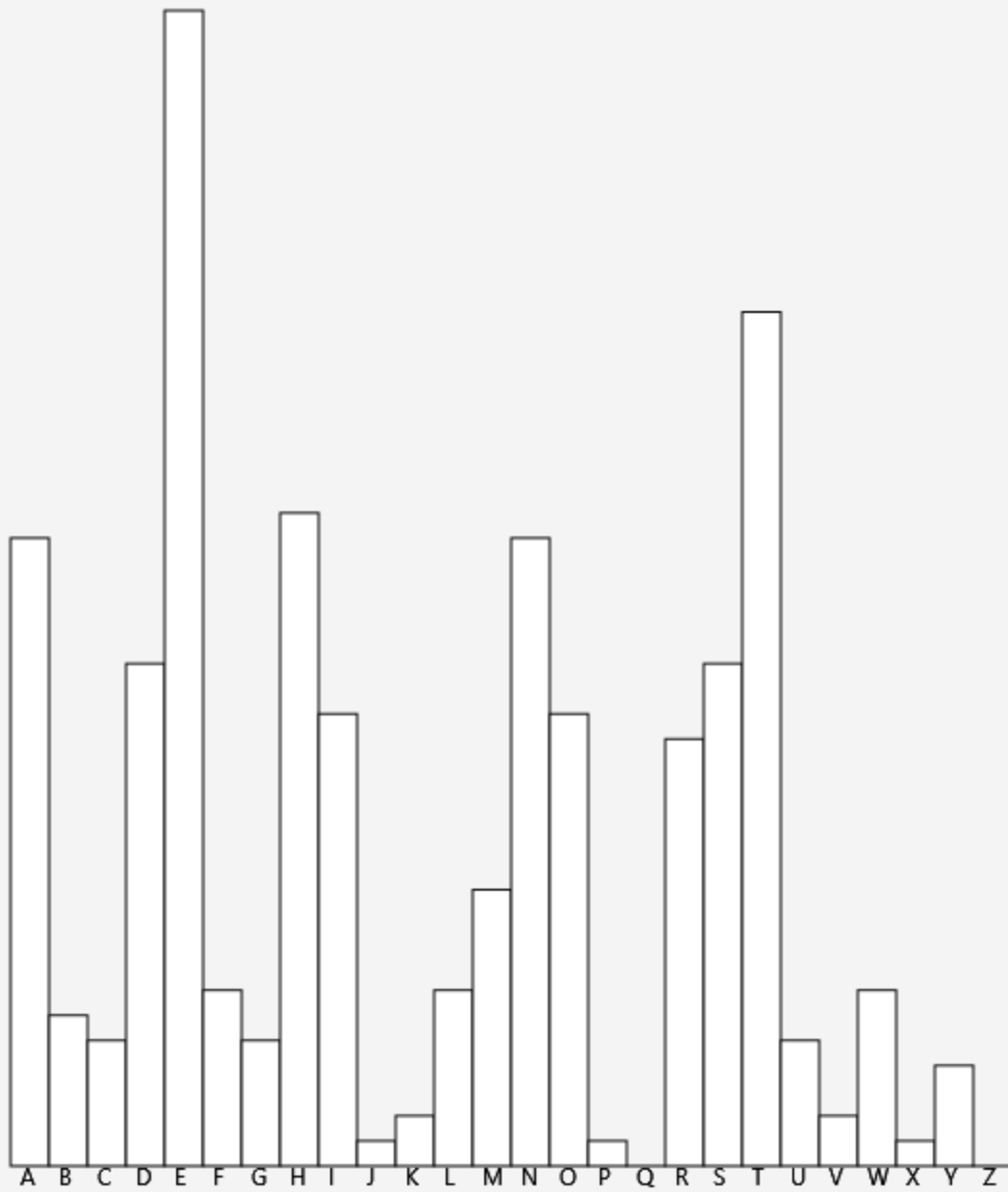
public Histogram() {
    this.finalCount = new int[26];
    maxValue = 0;
    generateHistogram();
}
}

```

Screen shots:

Include two screen shots here:

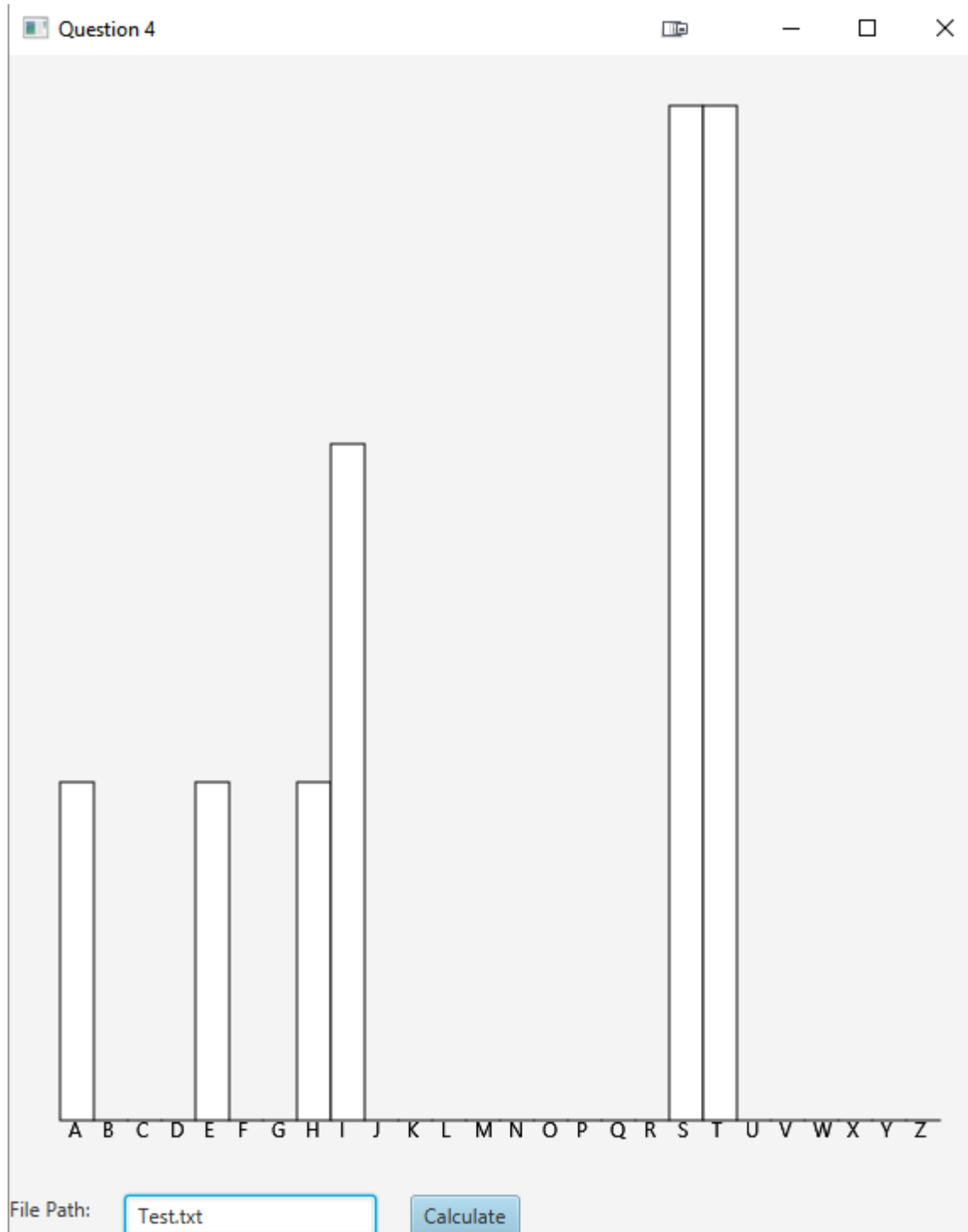
Question 4



File Path:

Test.txt

Calculate



Remember:

You need to complete this file and submit it in related **drop box on Blackboard**, in addition to uploading your codes in your **git repository**, before deadline.

<https://github.com/SchereChaSiu/CSCI2020U-Assignment.git>