Object Orientation – Spring 2018 **Assignment 12**

JavaFX Drawing Program

This assignment should be made at the "Online test" in HAL 1 of the Elinor Ostrom building at Friday May 18, 8:30-11:30. This should give you (and us) an impression of how the digital exam configuration works with the NetBeans IDE. In the afternoon your results will be available online. Finish the assignment and hand it in via Blackboard as usual. The work in the exam environment will not be graded!

1 Learning Goals

Having completed this assignment, you should be able to:

- make a more complex GUI in JavaFX;
- implement a mouse handler;
- place GUI objects at a computed position on the stage;
- dynamically move GUI objects.

This assignment is a direct translation from an assignment in one of the exams of last year. This assignment contributed 20% of the final mark of that exam. It should give you a good fairly good impression of the level of the exam and what you should know by heart of JavaFX programming. The appendix of this assignment contains the relevant part of the appendix of that exam.

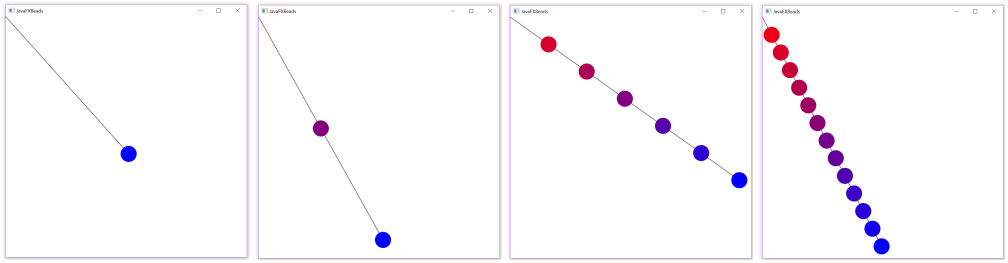
Problem Sketch: A String of Beads

Write a JavaFX program that makes a string of beads. The program starts with a single Blue bead somewhere on the screen. You can choose the initial position freely. This bead is always connected by a black line to the left-upper corner of the screen. When the mouse enters the screen the centre of this blue bead becomes the position of the mouse. The bead stays centered at the mouse while the mouse is moved inside the screen.

A new beads is added to the line when the mouse is clicked. The beads will distribute themselves evenly over the line. The colors of every bead is different. The bead at the mouse is blue. The color of other beads should gradually change from blue to red according to their position on the string. A bead near the left-upper corner should be almost red. Beads between those extremes have their own variant of purple.

Hint Use the static method `Color rgb( int r, int g, int b )`. Blue is `Color.rgb( 0, 0, 255 )` and red is `Color.rgb( 255, 0, 0 )`. The green component is always 0 for our beads.

The figures show the initial situation and the view after 1, 5 and 12 mouse clicks. The mouse is moved between those clicks.



2 Hand in

Before Monday, May 21th, 23:59, on Blackboard.

Running NetBeans in the Exam Environment

The exams run on a Chromebook. This Chromebook will run just two programs:

1. The safe exam browser, SEB, which contains the exam questions and will store your answers.
2. A remote desktop to a virtual windows machine running Windows. NetBeans runs on this virtual machine. This machine has very limited connections to world, hence you cannot use a search engine like Google.

Any work on the remote windows machine will be lost after the exam. You can use the IDE to prepare your answer and copy-past it to the SEB. Since the SEB and the virtual machine are different 'worlds' copying is somewhat complicated.

Copy from IDE to SEB Select the text to be copied from the IDE and press Ctrl+C. Select the 4th button from the left at the top of your screen and click Copy to local device. Press Ctrl+C again. Go to answer box in TAO and press Ctrl+V. Your text should now be copied.

Copy from SEB to IDE Select text to be copied in TAO and press Ctrl+C. Go to the editor and select the 4th icon from the left at the top of your screen and click Paste to remote session. In the editor menu select paste from the dropdown list. Your text should now be copied. Unfortunately, Ctrl+V does not work here.

Java en JavaFX Classes

A selection of Classes and methods you might need in this assignment.

```
public interface EventHandler<T extends Event> { void handle(T event); }
public class Circle {
    public Circle();
    public Circle(double radius);
    public Circle(double radius, Paint fill);
    public Circle(double centerX, double centerY, double radius);
    public void setFill(Color c);
    public void setCenterX(double value);        // idem for Y
    public DoubleProperty centerXProperty(); }  // idem for Y
public class Button {
    public Button(String text)
    public void setOnAction(EventHandler<ActionEvent> value); }
public class TextField {
    public TextField(String text);
    public String getText();
    public void setText(String text); }
class Scene {
    public Scene(Parent root, double width, double height);
    public double getWidth();
    public ReadOnlyDoubleProperty widthProperty();
    public double getHeight();
    public ReadOnlyDoubleProperty heightProperty();
    public DoubleProperty prefWidthProperty(); }
public class Stage {
    public Stage();
    public void setTitle(String title);
    public void setScene(Scene value);
    public void show();
    public void close(); }
```