Langara
THE COLLEGE OF HIGHER LEARNING.

Department of Computing Science & Information Systems
CPSC 1181
Lab#6
Oct 20, 2022

Objectives:

to learn to design a graphics application using an object oriented approach
to write code that is easily modifiable and expandable
to have fun with a graphics application

Preparation:

Study class notes and exercises.
Study slides 50 to 55

Due date:

Due Date: 11:00 PM on Wednesday Oct 26, 2021

Where to upload:

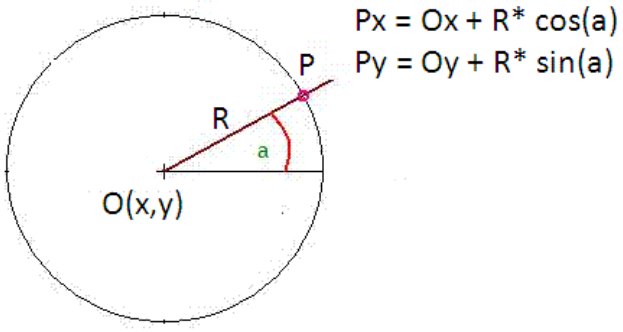
Zip your files into yourstudentID.zip where yourstudentID is your student number, and upload it to dropbox lab3 in D2L.

Part A: Tutorial

Tutorial to this part is added to the end of this lab document as example. Refer to the end of the document, practice and learn it. You do not need to upload anything for this tutorial.

Part B:

Consider Circle O with center at coordinate x and y, and radius R.



Px = Ox + R* cos(a)
Py = Oy + R* sin(a)

The coordinates of every point, P, on the perimeter of the circle is calculated as shown below:

Px = Ox + R * cos(a)
Py = Oy + R * sin(a)

where a is the degree R makes with horizontal line.
For example if O, center of the circle, located at (100,200) and R is equal to 170 pixels, then the coordinates of Px and Py are calculated as

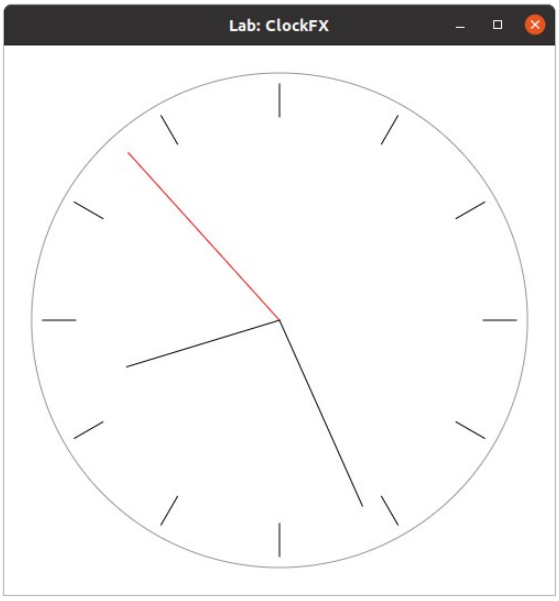
Px = 100 + 170 * Math.cos(a)
py = 200 + 170 * Math.sin(a)

where a is in radian.

Refer to an example at the end of the document.

Phase I: [50 marks]

Design and implement class Clock that displays the current time (the moment the program starts). Note that this is an OOP (Object Oriented Programming), and you should use JavaFX feature we have covered in this course.



The above figure indicates that the time the program start running was 08:26:53. **Note the position of the hour handle.**

Specifications:

- Set size of the frame to 500x500
- The clock is drawn in the middle of the scene, and its diameter is equal to 450 pixels.
- The outer circle boarder should be drawn using a gray color (choose the gray level that best suites your drawing).
- The length of the hour-handle should be smaller that the length of the minute-handle, and the length of the minute-handle should be smaller that the length of the second-handle (choose lengths that best suited to your drawing).
- Use a different color to indicate the second-handle.
- You should use the features we have covered in this course to draw the clock, otherwise, your solution will not be marked.

A clock like the one shown above is the minimum required for this assignment.

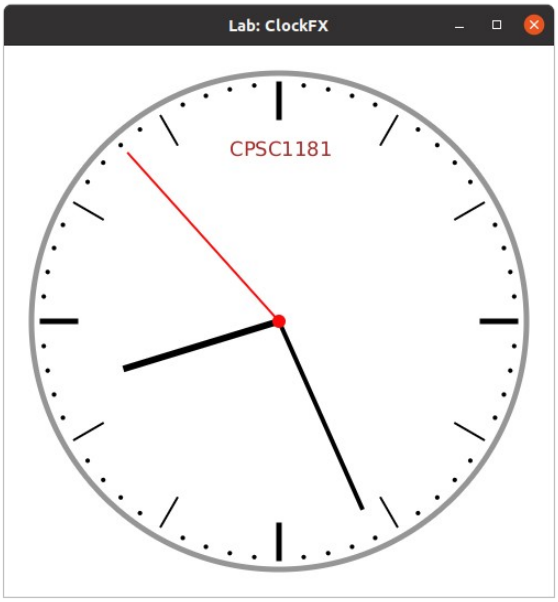
Important Note:

Note that this assignments it not learning graphics. It is part of the course to learn how to use objects, manage them, and use the Java library properly.

So, please do not search for solutions online. You are expected to study your text book, notes, and examples provided to start and end the assignment, otherwise, you will not learn what you are supposed to learn in this course.

Phase II: [15 marks]

Modify the Clock class that it looks the clock shown below:



The above figure indicates that the time the program start running was 08:26:53. Note the position of the hour handle.

Specifications:

- The outer circle boarder width of the clock should be set to 5 pixels.
- Use line-width equal to 2 pixel to indicate every 5 minute on the face of the clock (choose line length that best suited to your design).
- The position of each minute/second of the clock should be indicated by a circle with diameter equal to 4 pixels on the face of the clock.
- Use line-width equal to 5 pixel to indicate every 15 minute on the face of the clock.
- The line-width to draw hour-handle, minute-handle, and second-handle should be set to 6, 4, and 2 pixels respectively.
- Draw a circle in the middle with the same color as the second-hand with horizontal and vertical diameter equal to 7 pixels.
- Add CPSC1181 to the clock and center align it. Refer to the end of the lab to see how you can do it.
  - Use Arial font size 18, and brown color for the text.

Important Note:

You should only use the classes we have covered in the lecture. Using features, objects, and other JavaFX classes and methods that are not covered in the notes, will result in Zero mark.

Notes:

1. Use LocalDateTime to get current time as shown below:

```
LocalDateTime now = LocalDateTime.now();
int hours = now.getHour()%12; // getHour() returns a number between 0 – 23
int minute = now.getMinute();
int seconds = now.getSecond();
```

2. Avoid using **deprecated** classes and deprecated methods.

Bonus: 5 marks

Save the clock as a FancyClock.java file, and then modify it to make your clock looks fancy.
You will get a mark from 0 to 5 based on your marker decision. Check google for images of clocks to get an idea.

Notes:
1. It is strongly recommended to keep a backup of your codes before modifying them in case something goes wrong.
2.You should modify the class Clock to make it fancy. Submitting different implementation will not be marked.

What to submit

1. Comment all your classes and methods using the javadoc notation.
2. Your FancyClock.java will be marked if you implement the main part, Clock.jaga.
3. Your source files (Clock.java and FancyClock.java, and all other Java source files you created).
4. Comments about your assignment if needed these comments are not the comments documenting your code but rather something you need to convey to us about your assignment
5. Submit to D2l Dropbox: lab6

TOTAL MARK: 65

Examples:

1. Drawing handles based on time

Calculating and drawing line from center (200, 200) in direction of 29 seconds:

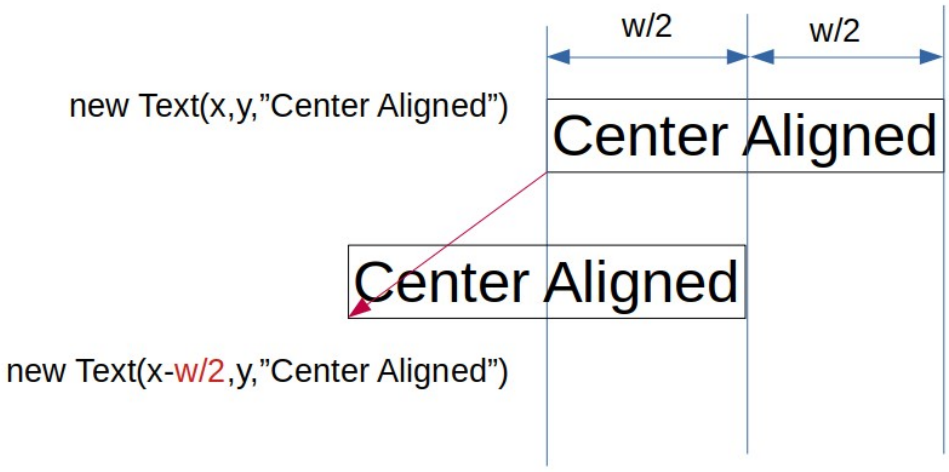
```
int second = 29; // position of 29 seconds
int r = 100;  // length of the line
int cx = 200;
int cy = 200;
// subtract 15 from seconds:
// since the zero angle starts from 15 seconds position
double angle = (second-15)*2*Math.PI/60;
double x2 = cx +  r * Math.cos(angle);
double y2 = cy +  r * Math.sin(angle);

Line secondHandle =  new Line(cx,cy,x2,y2);
```

2. Center-aling Text

FavaFX provides different functionality to align text ,however, we have not covered them yet. We will cover few of them in the coming lectures, and the rest will left for curious students to explore.
For this lab assignment, you can use following approach to find the coordinate of your text on screen to align it on the middle.

To align the text to the center, just shift the base half of the width of the text to the left.



Following example shows how to center align a text horizontally around coordinates (100,100)

```
....

import javafx.geometry.Bounds;
import javafx.scene.shape.Shape;
....


int x=100;
int y=100;
Font font = Font.font("Times New Roman", FontWeight.BOLD ,15);
Text tmp= new Text("Center Aligned"); // create a temporary text
tmp.setFont(font);

Bounds bound = tmp.getBoundsInLocal();
Rectangle box = new Rectangle(bound.getMinX(), bound.getMinY(), bound.getWidth(),
        bound.getHeight());
Shape intersection = Shape.intersect(tmp, box);
Bounds boundingBox = intersection.getBoundsInLocal();
double width = boundingBox.getWidth(); // returns width of the bounding box
//double height = boundingBox.getHeight();// returns height of the bounding box
// Now your text will be center aligned
test = new Text(x-width/2, y, "Center Aligned");
test.setFont(font);
```

Then add the text to a Panel.