Langara
THE COLLEGE OF HIGHER LEARNING.

Department of Computing Science & Information Systems
CPSC 1181
Lab#3
May 29, 2019

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 exercised
Study sections 2.9, 2.10, and 3.8 from your text book.

Due date:

Due Date: 11:00 PM on Tuesday June 4, 2019

Where to upload:

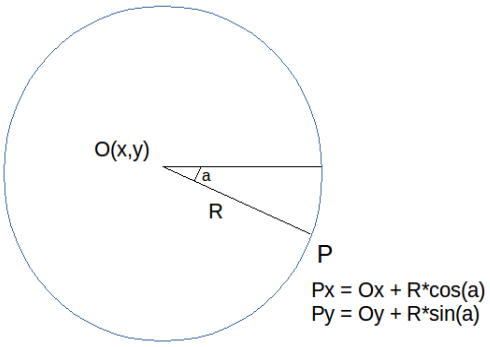
Zip your files into yourstudentID.zip where yourstudentID is your student number, and upload it to dropbox lab3 in D2L.

Part A: Tutorial

Do tutorial3
You should finish tutorial and upload your files by 17:00 today.

Part B:

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



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.

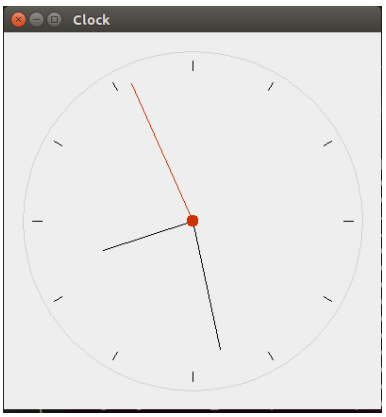
For example if O, center of the circle, located at (100,200) and R is equal to 170 pixels, then the coordinates of Px and Py are calculated as

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

where a is in radian.

Phase I: (80% of the mark)

Design and implement class Clock that displays the current time (the moment the program starts). . Note that this is OOP (Object Oriented Programming), and the Clock class must have an instance method so that the clock can draw itself on a JComponent using the Graphics2D.



The above figure indicates that the time the program start running was 08:27:56. Note the position of the hour handle.

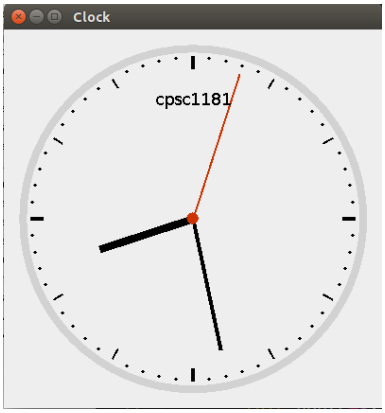
Specifications:

- Set size of the frame to 400x400
- The clock is drawn in the middle of the frame, and its diameter is equal to 360 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.
- Note there is another small red circle (the same color as the second-handle) in the center of the clock.
- Do not animate the clock, and do not use thread to implement it (We have not covered them yet).

A clock like the one shown above is the minimum required for this assignment. If you draw such a clock, you will receive at most 80% in the programming part of the assignment.

Phase II: (20% of the mark)

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



The above figure indicates that the time the program start running was 08:28:03. Note the position of

the hour handle.

Specifications:

- The outer circle boarder width of the clock should be set to 8 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 4 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 8, 4, and 2 pixels respectively.
- Use Arial font with size equal to 18 to display CPSC1181 on the face of the clock. Note that center aligning the message “CPSC1181” horizontally is a challenge. I suggest use try and error, thought there is a better way to do it, but it is not part of our course.

To finish your lab assignment , you need the following classes:
java.awt.Graphics;
java.awt.Graphics2D;
java.awt.Color;
java.awt.Font;
java.awt.BasicStroke;
java.awt.geom.Ellipse2D;
java.awt.geom.Line2D;
javax.swing.JComponent;
javax.swing.JFrame;
java.time.LocalDateTime;

Notes:

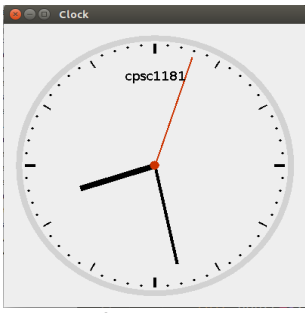
1. Use LocalDateTime to get current time as shown below:

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

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

Bonus_1: 5 marks

Make your clock grows or shrink by resizing your frame. Note than your clock should always be drawn in the middle of the frame. The following two figures shows the output of the program before and after resizing the frame.



Frame size: 400 x 400

Frame size: 860x600

Bonus_2: 5 marks

Create class FancyClock that looks like a fancy clock. You will get a mark from 0 to 5 based on your marker decision. Check google for images of clocks to get an idea.

Note:

It is strongly recommended to keep a backup of your codes before modifying them in case something goes wrong.

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: lab3

TOTAL MARK: 60