

Using Predefined Objects

Out: 2/3

Due: 2/12 by 11:50 PM

Maximum Score Possible: 25 points

In this project you will write a program that constructs two abstract ellipses, translates and resizes them. The program will also compute the perimeters and areas of both ellipses.

Learning Objectives

- Writing, compiling and running a simple java program
- Using Objects
- Invoking class and instance methods
- Importing a package
- Navigating the Java API documentation

Relevant Formulas

Let w denote the width and h denote the height of an ellipse then the formulas for the *area* and *perimeter* of an ellipse are given by:

$$area = \frac{\pi wh}{4} \tag{1}$$

$$perimeter \approx \frac{\pi}{2} \left[3(w + h) - \sqrt{3w^2 + 10wh + 3h^2} \right] \tag{2}$$

1 Working With Abstract Ellipses

In class we illustrate how to create and use objects of a class by exploring the online API (Application Programmers' Interface) documentation of the class. We illustrated this by using the `Rectangle` class defined in the `java.awt` (Abstract Windowing Toolkit) package. In this project, you will create and use objects of the `Ellipse2D.Double` class. An ellipse specifies an area in a coordinate space that is enclosed by an `Ellipse2D.Double` object whose framing box

has its top-left point (x, y) in the coordinate space, its width, and its height. Navigate your way to the Java API documentation for the `Ellipse2D.Double` class available at <http://docs.oracle.com/javase/7/docs/api/> to familiarize yourself with all the methods available in this class. Using the class, write a program, `EllipseTester.java`, to do the following:

1. Construct an ellipse whose framing box has its top-left point coordinates are $(60, 40)$ and whose width is 120 and height is 80.
2. Construct a smaller concentric circle with the parameters to the constructor defined relative to those of the first triangle. The height and width of the circle should be the same as the height of the ellipse. (Note: Arguments to the constructor should be a expressions rather than constants.)

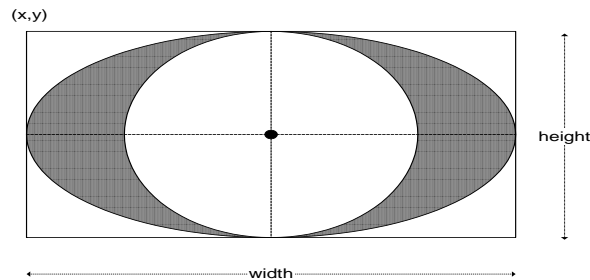


Figure 1: A Circle Inscribed in an Ellipse

3. Using appropriate methods, print the coordinates of the vertices of the circle and the ellipse in the form $\{(c_x, c_y), r_1, r_2\}$, where the (c_x, c_y) coordinates represent the center of each object and r_1 and r_2 represents the width and height, respectively, of each object. Use a label to indicate which object's coordinates are printed. Also, using appropriate methods, calculate and print the area of the shaded region between the two objects (Hint: subtract the areas of the circle from the ellipse.)
4. Translate both objects to the right along the x coordinate axis by 10, and upward along the y coordinate axis by 10.
5. Again, print the coordinates of both ellipses using appropriate labels and the format described in 2. Also, print the area of the shaded region.
6. Now, use an appropriate mutator method in the ellipse class to increase the width by 10 and height by 20 of the ellipse. Also, change the width and height of the circle so that its width is half the width the ellipse and its height is half the height of the ellipse.

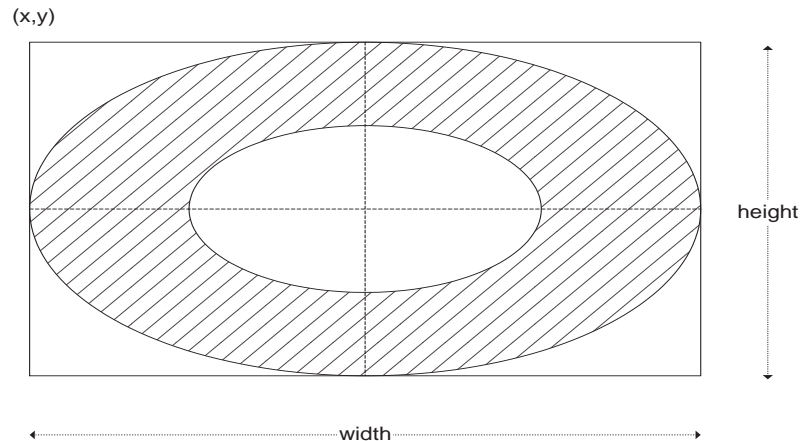


Figure 2: An Ellipse Inscribed in another Ellipse

(Read and navigate your way through the API documentation to find a mutator method that does this. Invoke this method twice using the two objects. Do not create another `Ellipse2D.Double` object to accomplish this.)

7. Again, print the coordinates of both ellipses using the format described above. Also, print the area of the shaded region between the two ellipses. Include appropriate labels.
8. Print the approximate perimeters of each ellipse. Print the results with appropriate labels.
9. Now, compile and run the program and note the area of the shaded region between the ellipses and their perimeters.
10. Finally, change only the top-left coordinates of the first ellipse to (45,55) and compile the program and execute the program. If you set up your code correctly, the perimeters of the ellipses and the area of the shaded regions between them should remain the same.

A typical program output would be:

```
The coordinates of the bigger ellipse are {(120.0,80.0),120.0,80.0}.
The coordinates of the smaller ellipse are {(120.0,80.0),80.0,80.0}.
The area of the shaded region is 2513.274122871834.
```

```
After the ellipses are translated:
The coordinates of the bigger ellipse are {(130.0,70.0),120.0,80.0}.
The coordinates of the smaller ellipse are {(130.0,70.0),80.0,80.0}.
The area of the shaded region is 2513.274122871834.
```

```
After the ellipses are resized:
The coordinates of the bigger ellipse are {(135.0,80.0),130.0,100.0}.
The coordinates of the smaller ellipse are {(135.0,80.0),65.0,50.0}.
The area of the shaded region is 7657.632093125119.
```

```
The approximate perimeter of the bigger ellipse is 362.8214411874936.
The approximate perimeter of the bigger ellipse is 181.4107205937468.
```

Name the file containing this program *EllipseTester.java*.

Submitting Your Work

Compile your Java source files (EllipseTester.java) until it is free of any syntax and/or semantic errors. You may submit your work as many times as you wish once the due time has not elapsed. After the due time, you will NOT be able to submit your work for grading. You will receive no points if your work is not submitted. Be sure to follow the coding style guidelines we discussed in class. Your program will be graded for both proper style and syntax.

1. The source code files you submit must have a header with the following:

```
/*
 * @file filename
 * @author your name
 * @date 9999-99-99
 * Description: describe what this program does
 * Course: CS1350.0x
 * PAWS ID: xxxxxx
 * Project #: 1
 * Instructor: Dr. Duncan
 */
```

2. Once you are convinced that your program works correctly, locate your source file (EllipseTester.java).
3. Create a zip file myPawsID_2.zip and add the file to it, where myPawsID is your LSU paws ID - the part of your LSU email address before the @ sign.
4. Submit your work (the zip file containing your source file) for grading via electronic drop box on Moodle.