Programming Exercise: Calculating a Shape's Perimeter

Note: You must have downloaded BlueJ on or after September 7, 2017, from http://www.dukelearntoprogram.com/ in order to access the Shape and Point Java classes used in the following programming assignments.

In these exercises, you will use the Shape and Point classes to answer questions about a Shape that is made up of a collection of points from the x-y plane, as shown in this lesson. The shape is defined by drawing a line between two adjacent points, for every pair of adjacent points, and also a line between the first and last point. Be sure to consult the documentation on DukeLearnToProgram to understand how the Shape, Point, DirectoryResource and FileResource classes work: http://www.dukelearntoprogram.com/course2/doc/javadoc/index.html?course=2.

Assignment 1: Calculating information about shapes

In this assignment, you will complete the PerimeterAssignmentRunner class to calculate lots of interesting facts about shapes. This class has been started for you in the BlueJ project called assignmentPerimeter (go to: http://www.dukelearntoprogram.com/course2/files.php and download the Calculating the Perimeter of a Shape BlueJ project). This project also contains several data files. In addition, you will need to look at the documentation for the Shape class and the Point class.

The PerimeterAssignmentRunner class already includes the following complete methods:

(a) The getPerimeter method has one parameter s of type Shape. Given a shape, this method returns the perimeter of the shape.

(b) The testPerimeter method has no return value, hence its return type is void. This method is used to select a data file by using the FileResource class, create a shape based on the points from that data file, and then calculate the perimeter of the shape and output its value.

(c) The triangle method has no return value and creates a triangle that you can use to test the methods you will create in this assignment.

(d) The printFileNames method, which we will discuss in Assignment 2.

(e) The main method.

For this assignment, you will add or modify several methods in the PerimeterAssigmentRunner class.

1. Complete writing the method getNumPoints that has one parameter s that is of type Shape. This method returns an integer that is the number of points in Shape s. Hint: You will need to iterate over all the points in the Shape S and count them.

2. Add code in the method testPerimeter to call getNumPoints and to print the result.

3. Complete writing the method getAverageLength that has one parameter s that is of type Shape. This method returns a number of type double that is the calculated average of all the sides' lengths in the Shape S.

4. Add code in the method testPerimeter to call the method getAverageLength and to print out the result. Note if you were to select the file example1.txt, then the average side length should be 4.0.

5. Complete writing the method getLargestSide that has one parameter s that is of type Shape. This method returns a number of type double that is the longest side in the Shape S.

6. Add code in the method testPerimeter to call the method getLargestSide and to print out the result. Note if you were to select the file example1.txt, then the longest side should be 5.0.

7. Complete writing the method getLargestX that has one parameter s that is of type Shape. This method returns a number of type double that is the largest x value over all the points in the Shape s.

8. Add code in the method testPerimeter to call the method getLargestX and to print out the result. Note if you were to select the file example1.txt, then the longest side should be 4.0.

Assignment 2: Processing multiple Shape files

In this assignment you will find the largest perimeter over several shapes by examining several files representing shapes, calculating the size of the largest perimeter and also the name of the file with the largest perimeter. You will add new methods to the PerimeterAssignmentRunner class.

The PerimeterAssignmentRunner class already includes the following method you should understand the following. The printFileNames method has no parameters and no return value, hence return type void. This method first creates a DirectoryResource. When this happens you are prompted to select a file or files. You can select a bunch of files together

by clicking on the name of one file, and then hold down the shift key and select a second file. All the files between the first and second file will be highlighted. The code then iterates over all the files you have selected using a for loop and the selectedFiles method, printing out the filename for each file.

For this assignment, you will add or modify several methods in the PerimeterAssigmentRunner class:

1. Complete writing the method getLargestPerimeterMultipleFiles that has no parameters. This method creates a DirectoryResource (so you can select multiple files) and then iterates over these files. For each File f, it converts the file into a FileResource with the line

FileResource fr = new FileResource(f);

Then it creates a Shape from the FileResource and calculates the shapes perimeter. What else does it need to do? It needs to return the the largest perimeter over all the shapes in the files you have selected.

2. Finish writing the void method testPerimeterMultipleFiles to call getLargestPerimeterMultipleFiles and to print out the largest such perimeter. This method has no parameters and no return value. You will select the files when you run this method (hint: see our documentation for the DirectoryResource class).

3. Finish writing the method getFileWithLargestPerimeter that has no parameters. This method should, like the getLargestPerimeterMultipleFiles method, create its own Directory Resource, except that this new method returns the File that has the largest such perimeter, so it has return type File.

4. Add code to the method testFileWithLargestPerimeter to call getFileWithLargestPerimeter. For the File that is returned, print the name of that file.

It is important to test your code with several files.

✓ Complete