CSC-240 Lesson 9: Graphics and Java 2D

Assignment 9

Please write the programs as described below. Please submit at least the following files to the assignment (you may provide additional files as needed to properly decompose the problem):

- `Grid.java`—the class that is the main program that sets up and displays a window as described below. Make use of further classes to help to structure the program, and those additional classes may have appropriate names (and files) of your choice.

- `Triangles.java`—the class that is the main program that sets up and displays the triangles as described below. Make use of further classes to help to structure the program, and those additional classes may have appropriate names (and files) of your choice.

Keep in mind that these programs are exercises in drawing. and that the only way to do drawing is programmatically (i.e., using drawing and graphics features).

Here are specific instructions and hints for each exercise.

Grids Using Method `strokeLine`

In this exercise you will use an appropriate class to draw a square grid within the window (which will require an instance of the **Canvas** class in JavaFX). For this program, follow the instructions below:

- The main program, `Grid.java`, should use the launch method of the Application class to create an instance of an Application class. The start method will be provided with a window (a **Stage** instance) which will eventually hold the grid pattern, and display it.

- From the application class, prompt the user to supply two values, one being a count of the number of cells across the grid. (The same number will be used for the number of cells down the grid, as it is to be square), the second being the (initial) size of each cell (in pixels). It is suggested that the method inputMessage of the MessageDisplay class provided in the package edu.frontrange.util.javafx.message with the TicTacToe example be used, and that the two values just be entered separated by a space.

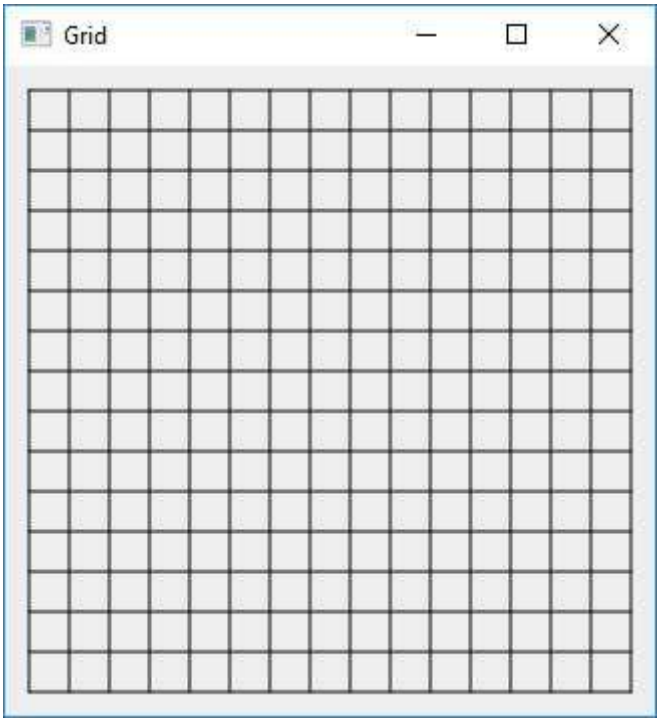
- If either of the two values are unacceptable (i.e., a value is negative, or less that a reasonable minimum value), then the user should be informed, and the value replaced by a pre-determined default value--do not bother with asking the user again. It is suggested that the method displayMessage of the MessageDisplay class provided in the package edu.frontrange.util.javafx.message with the TicTacToe example be used to show error messages to the user.

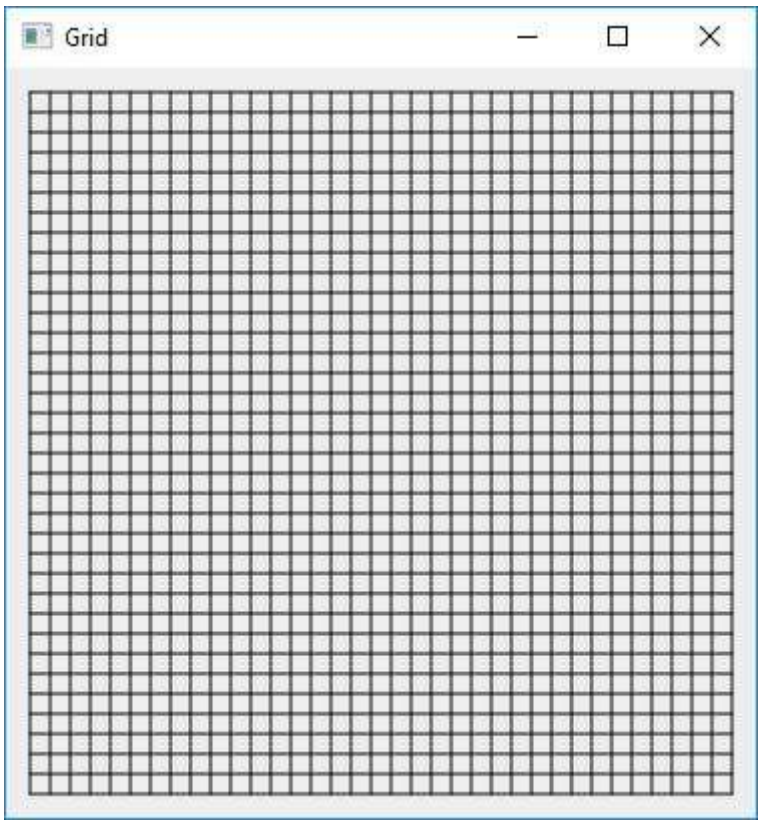
- Use these argument values or the replacement defaults to control the drawing of the grid. Use the method *strokeline* to draw the grid. Attempt to leave some amount of margin around the grid so that the edges may be easily seen.

- When the window is resized by the user (using the mouse), the grid pattern in the window should expand (or contract), while still remaining square, to continue to use as much of the window as possible, but keeping the ***same number of rows and columns***. That is, only the size of the cell, equivalently, the spacing of the grid lines, should change.

- The window must have a title.

This will give more practice in writing effective classes. Here are images of some windows that a working solution to this problem might display (and they may be anywhere on the screen):





The first grid was created with values: cell count 20, cell size, 20.

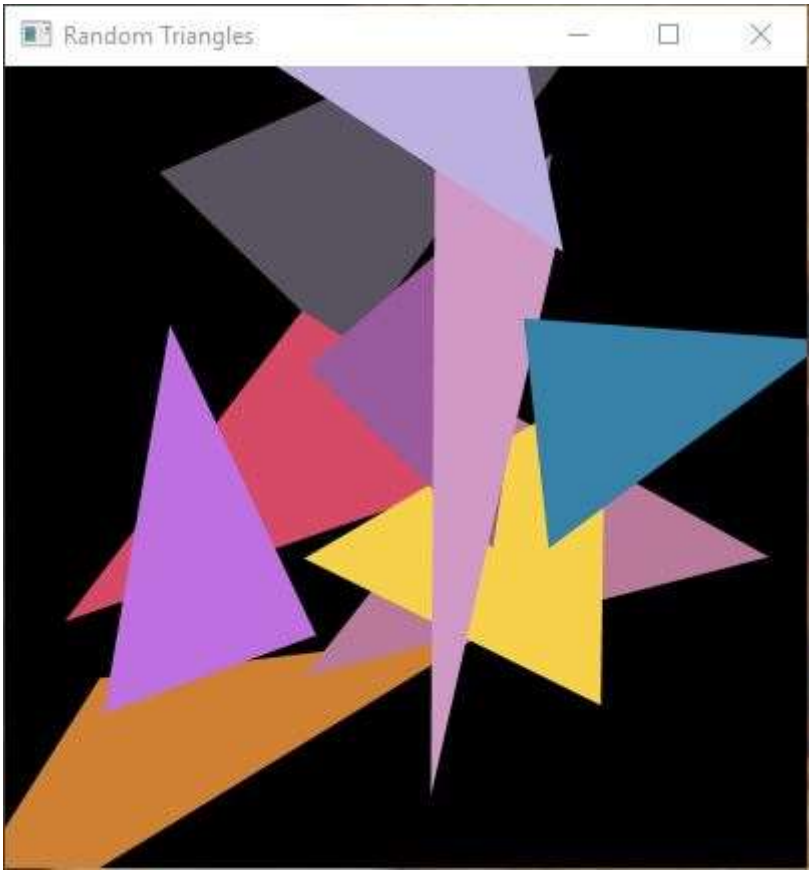
The second grid was created with values: cell count 35, cell size, 10.

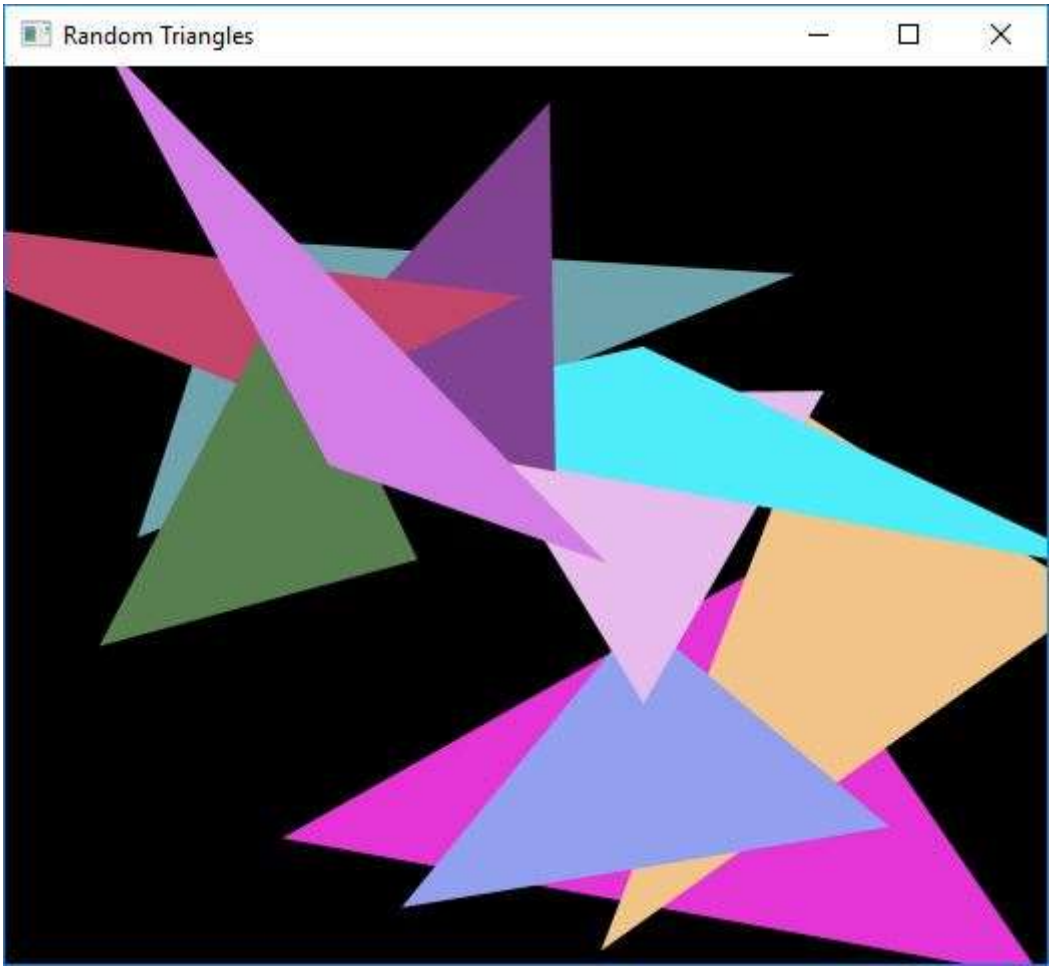
Random Triangles

Do this exercise as directed:

- Draw at least ten different triangles within a window.
- All the triangles should not only be of a random shape, but be filled in by a randomly selected color, and the display area itself should have a contrasting background color. All the triangles should have most or all of their shape inside the window.
- If the window is resized (*e.g.*, by grabbing the edge with the mouse and dragging the border of the window), the triangles should be redrawn with new triangles and colors, to fill the new size and shape of the window.
- Put an appropriate title on the application window.
- A **Random** object will be needed to generate the random values that you use to create the three points of the triangle that you will use to build the triangle objects, and to generate the random numbers that can be used to create colors.

Here are two examples of the solution in action. The images are from before and after dragging the edge with the mouse. You will notice how the application has been redrawn and/or resized whenever the window needs to be repainted (*e.g.*, if the window size is changed by the user dragging the edges or the corners). You will learn about how modern operating systems redraw windows, drawing only those portions that it must.





The following *About the Assignment* topic contains further explanation, directions, and hints about this assignment. Please read that topic before starting this assignment.

Important

Please follow the directions of Programming Assignment Identification in submitting your programming solutions.

If you have any questions or concerns about the Assignment or the *About the Assignment* topic, please use the Lesson Question discussion topic, or send a message by the D2L Internal Messaging system if you prefer.

Note: you may submit to the Assignment Submission Folders as many times and as often as you wish, up to the deadline time. Each submission is tagged with the date/time, and so each submission remains separate and distinct. Unless you leave instructions to the contrary, only the most recent of each file with the same name will be viewed for the purposes of grading. Details may be found in the topic *How To Submit and Get Feedback on Assignments* in the *How To* module.

Messages that accompany Assignment Submissions are read, and responded to, ***only*** when assignment submissions are graded (which is after the Assignment Submission Folder closing date/time). If you have a comment or question about an assignment, or a request for assistance, that needs an earlier response, then that comment, question or request should be made or asked *via* an Internal Message or the Discussion board, as these are usually read and answered every day.