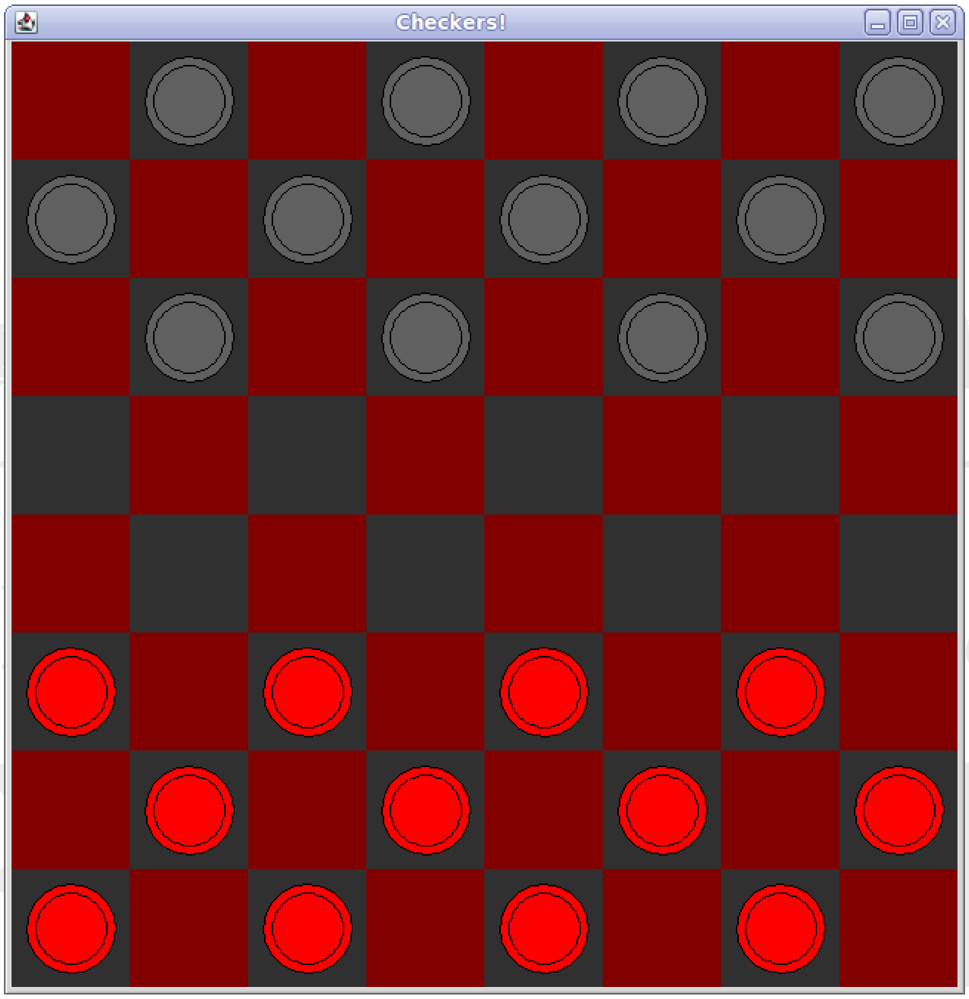
Lab 11: Drawing a Checkers Board

Draw a checkers board (called "draughts" in many countries), ready for play. The final image should look like this:



You should follow these principles:

- The board will be checkered, with a "light color" and a "dark color", and the lower-left square will be dark. The board's dimensions are 8 squares by 8 squares. (Traditionally, these colors are red and black—though you may use any colors you wish so long as light and dark are easily distinguished.)
- Pieces are also light and dark (again traditionally red and black), and can only be placed on dark squares. The top three rows will be filled with dark (black) pieces, and the bottom three rows will be filled with light (red) pieces.
- A piece's graphic will be a filled circle of the proper color, with two empty black circles drawn on top of it: one around the edge of the piece, and one slightly smaller and centered. The piece should be easily visible against its dark background. Its height and width should be three quarters the height and width of a square, and it should

be centered. (You will probably have to use both the `fillOval()` and `drawOval()` methods to draw them properly.)

- The default size of the board will be 640×640 pixels, and this value should be stored in a final variable. However, the programmer should be able to change this size and recompile, and the entire board and all its pieces will grow and shrink proportionately.

Try to think through the functions that you will make. Writing the correct functions can make your job much easier. For example, traditional checkers numbers the black squares from 1 to 32, starting in the upper left and going across. One idea is to make a `drawChecker()` function that takes an int and a `Color` object, and draws a piece of that color in the proper square. This might be a little hard to write, but once you have it, you have a function that you can use over and over again with ease!

This class will be called `Checkers`.