Practice exercises #5-1

# Run-time analysis of the Game of Life algorithm

Let g = number of generations

Let n = width of the grid (assume the grid is a square)

In this exercise, you will show that the run-time T(n, g) for the Game of Life algorithm is dominated by the factor gn2.

Use the file GameOfLifeSOLUTION.java in your Unit 5 owl folder.

**Step 1**: Write the condensed code for computeNextGeneration. Then write a simple expression for Tcng(n), the run-time for computeNextGeneration.

**Step 2**: Do the same for **paint**, **plantNextGeneration** and **makeEveryoneDead.**   
Let Tp(n), Tpng(n), and Tmed(n) represent their worst-case run-times.

**Step 3:** What do all four of these T functions have in common?

**Step 4:**  Let T(n, g) represent the total run-time for the entire GOL simulation. Write a simple expression for T(n, g) in terms of Tcng(n), Tp(n), Tpng(n), Tmed(n) and g.

**Step 5:**  Substitute the expressions for Tcng(n), Tp(n), Tpng(n), and Tmed(n) into the expression for T(n, g) and simplify. Show that the dominant term is some constant times gn2.

# Pick any Java program you’ve written this term that has at least one loop.

Do a run-time analysis for that program. For example, you could take the Star Wars character program.