COMP210 - Project 2 - Conway's Game of Life
Spring 2014

John Conway's *Game of Life*[1] is a classic programming problem and your second and final project for this course.

The Game

Game of Life is very well documented on its wikipedia page and we've discussed it in some detail in class. The game is built from a two dimensional universe of cells that live or die each turn based on a very simple set of rules. While the game might seem very simple, it is actual a Turing-Equivalent model of computation. This means that with the right population of cells and the right variation of the rules, Game of Life can compute anything that modern computers can compute. This means you could, in theory, implement your Game of Life project using the Game of Life. Don't worry, you don't have to rebuild the entire corpus of computing with your final project; we'll stick to Java.

Your Game

The minimum requirements for your implementation of Game of Life are:

- Maximize population size while maintaining a playable game.

- Players should be able to stop and start the game whenever they like.

- While the game is stopped, players should be able to modify the current population of cells.

- While the game is going, the game should display the number of iterations since the last stop or since the last player intervention with the population. So, if the player stops and starts the game without killing or resurrecting any cells, then the count should not change.

In addition to the minimal functionality, you are expected to add one or more features that enhance the game. Some possibilities are:

- The ability to add predefined patterns like gliders

- The ability to set the initial population from a text file.

- Random populations.

- Mutation: cells have a non-zero probability of switching states regardless of the state of their neighbors. This changes the game, so you should probably the user to turn it on/off in case they want the traditional deterministic game.

- The ability to zoom in and out on portions of the population. This is likely to include scrolling while viewing the population as well.

- Visually engaging and usable UI. Consider adding things like help text and in-game guides.

Evaluation

Your project will be evaluated on the following criteria:

1. Utilizing Good MVC based design

2. Robust unit testing

3. Coding style, documentation, and human readability.

4. Effective use of Java Libraries

5. Meeting the minimum requirements

6. Exceeding the minimum requirements

A moderately well done program that completely meets the minimum requirements can expect a passing grade in the C range. A superbly well done program that meets the minimum requirements can expect something in the C+/B- range. Grades beyond that level will require enhancements to the basic game described above.

Dates

- Lab on 4/17 and 4/24 will be dedicated to project work.

- Complete set of unit-tests due by class time **Wednesday 4/23**. Submit as **proj2quiz**

- **Due on Monday 4/28. Submit as *proj2* via handin**