PROJECT 6
Conway's *Game of Life*

1 A rant

Attendance during Friday's lab sections has become unacceptably poor. I presented this assignment in lab without posting a description, and consequently, the number of people familiar with the project was few. You are expected to be in lab, and you are responsible for the material presented during lab sections. I will not re-teach material to those who are not in class and in the lab during our class meetings.[1]

2 Your assignment

Implement Conway's Game of Life. This assignment will be the first in which you write a program from scratch. To get started, login to `remus/romulus` and do the following:

```
$ mkdir project-6
$ cd project-6
$ cp ~sfkaplan/public/COSC-111/project-6/* .
$ emacs Life.java &
```

The `cp` command will provide two pairs of files:

- `simple.init` and `simple.results`: A file that describes a small, initial configuration for a Game of Life, and then a file that shows the desired output from playing out that initial configuration.

- `X-pattern.init` and `X-pattern.results`: A slightly larger and more complex initial configuration, along with the output that the configuration yields over a few generations.

In your new `Life.java` file, you should write a program that:

1. Reads an initial file via `stdin` (that is, as though it were entered through the keyboard) and create a two-dimensional array that represents that initial grid.

2. Prints the initial grid, matching the format shown in the `.results` files.

3. Evolves the grid, setting each cell as being alive or dead based on its state and the states of its neighbors.

4. Prints each evolved grid.

5. Repeats the evolution-printing cycle until the grid reaches *stasis*—the cells do not change from one step to the next.

---

[1]Illnesses and emergencies are, of course, exceptions.

3 How to submit your work

Use the `cs111-submit` command:

```
$ cs111-submit project-6 Life.java
```

Project 6 is due on Thursday, April 25, at 11:59 pm