

# INTRODUCTION TO COMPUTER SCIENCE II

## LAB 4

### Abstracting Cells

## 1 Different cell types

Revisit the Game of Life page on Wikipedia, and read the section on *Variations*. Specifically, this passage describes that the standard cell in this game—what we will call a *Conway* cell—follows rules that could be varied. It describes a *Highlife* cell that is a modest variation on the *Conway*. A *Highlife* cell, when evolving, is...

- **born**<sup>1</sup> if it has 3 or 6 live neighbors;
- **survives**<sup>2</sup> if it has 2 or 3 live neighbors; and,
- **is dead** under all other circumstances.

## 2 Modifying your code

Create a new `lab-4` directory. Grab some new starting source code:

- **On `remus/romulus`**, perform the following command:  

```
$ cp ~sfkaplan/public/COSC-112/lab-4/*.java .
```
- **On your own computer**, download and extract this zip file.

You will find most of the same classes from Project-1. However, you must copy your own `Cell.java` code into this directory. If you examine `Life.java`, `Game.java`, and `Grid.java`, you will notice that a `cellType` variable is passed in the process of creating the grid of cells.

**What you must do.** Abstract the `Cell` class. That is, `Cell` must become an *abstract class*, with two *abstract methods*: `evolve()` and `toString()`. Then create two subclasses: `ConwayCell` and `HighlifeCell()`, which implement those two methods using their respective rules. (Use `+` and `-` to represent live and dead Conway cells; use `*` and `~` to represent live and dead Highlife cells.)

Finally, you must write a *static method*, `create()`, in the `Cell` class that can create a specific subclass object depending on the `cellType` string passed to it. More on this method in lab.

---

<sup>1</sup>That is, it is currently *dead* and becomes *alive*.

<sup>2</sup>That is, it is current *alive* and remains so.

### 3 How to submit your work

Use the CS submission systems to submit your `Cell.java`, `ConwayCell.java`, and `HighlifeCell.java` files. You may use either of the following two methods, while connected to `remus` or `romulus`, to use the submission system:

- **Web-based:** Visit the submission system web page.
- **Command-line based:** Use the `cssubmit` command at your shell prompt.

**This assignment is due on Sunday, Mar-05, 11:59 pm.**