Introduction to Computer Science II
Lab 3
Abstracting `Cell`

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**[1] if it has 3 or 6 live neighbors;
- **survives**[2] if it has 2 or 3 live neighbors; and,
- **is dead** under all other circumstances.

2 Modifying your code

Get the code: Use the following link to download a zip file with a bunch of source code:

`https://bit.ly/AMHCS-2020S-112-l3`

You will find most of the same classes from Project-1. 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. Just as importantly, notice that the `Cell` class is an *abstract* class—a concept that we will discuss at the beginning of lab. It has two methods, `evolve()` and `toString()`, that are *declared but not defined*: they have no body. Those methods must be defined in any subclass.

Take special notice, in the `Cell` class, of the method, `create()`. This static is responsible for creating a `ConwayCell` or a `HighlifeCell` depending on the cell type requested in the string `cellType`. This kind of creation method encapsulates the problem of turning an input request string into a specific subclass object type—something constructors cannot do.[3]

What you must do: Create two `Cell` 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.) Your `ConwayCell` class can use code you wrote from Project-1.

[1] That is, it is currently *dead* and becomes *alive*.
[2] That is, it is current *alive* and remains so.
[3] Do you see why not?

3 How to submit your work

Submit your `ConwayCell.java`, and `HighlifeCell.java` files via the CS submission system:

`https://www.cs.amherst.edu/submit`

This assignment is due on Sunday, Mar-01, 11:59 pm.