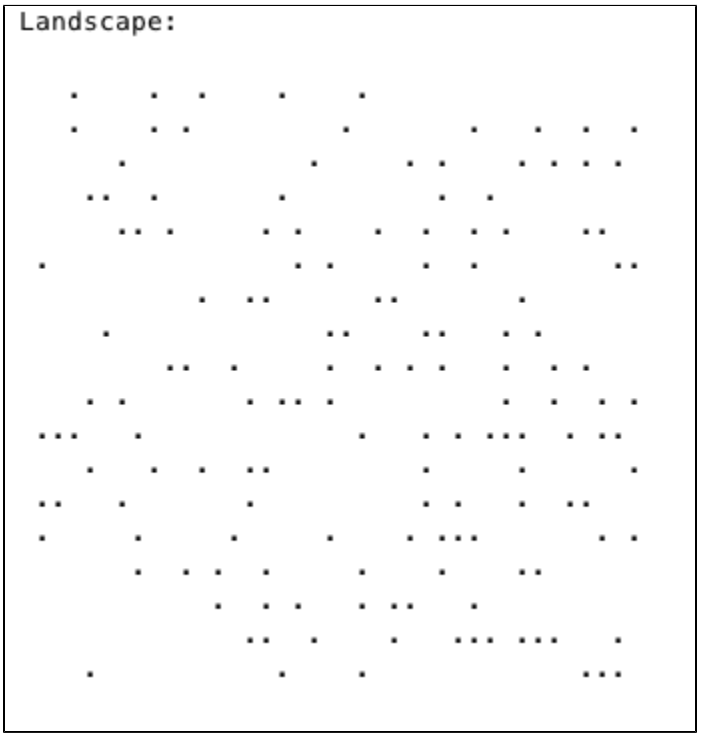
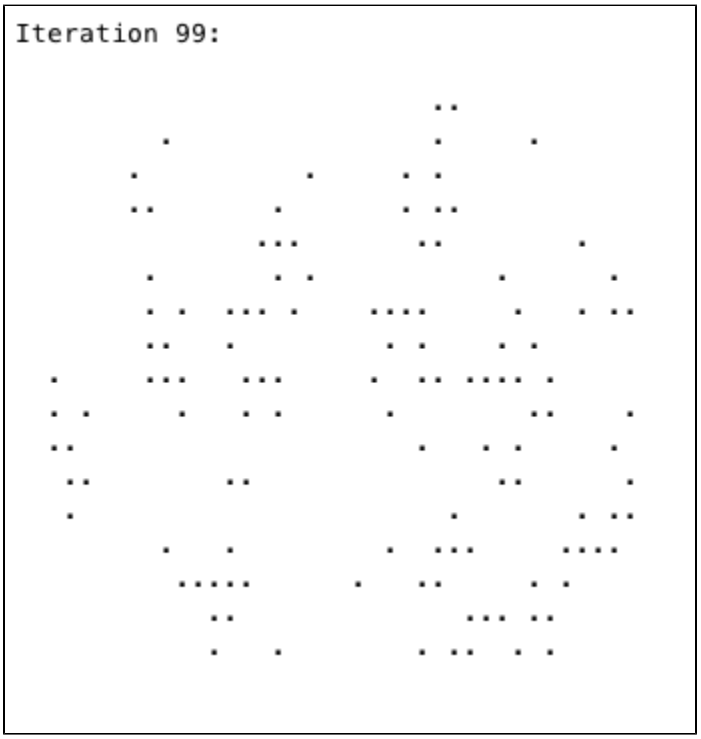
Siyuan CS231 Project3

This week's projects is modeling cell behaviors with different rules. The cell keeps track of its own location, and landscape uses the LinkedList class to track down the agents.

The Cell class returns position (x, y) as either decimals or nearest integers, returns a single period in its toString method, and sets up update state rules. It models a clumping behavior using the following rule: If the cell has more than 3 neighbors, then the cell should move +/- 5 with a 1% chance; else the cell should move +/- 5.

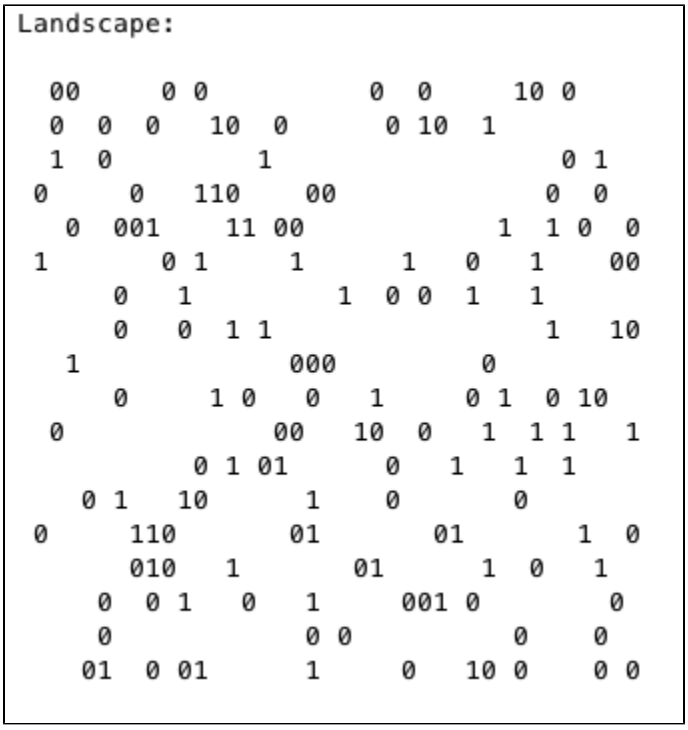


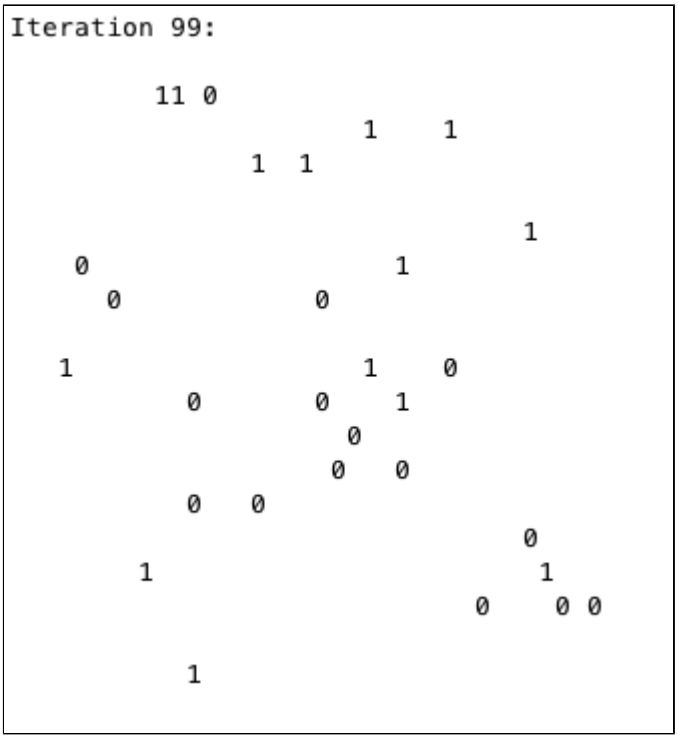


The Landscape class contains fields for the width and height and a LinkedList of Cell objects. It has methods to clear the Landscape of agents, return the height and width of the Landscape, add an agent, and return an ArrayList of the Cells on the Landscape. Additionally, the toString

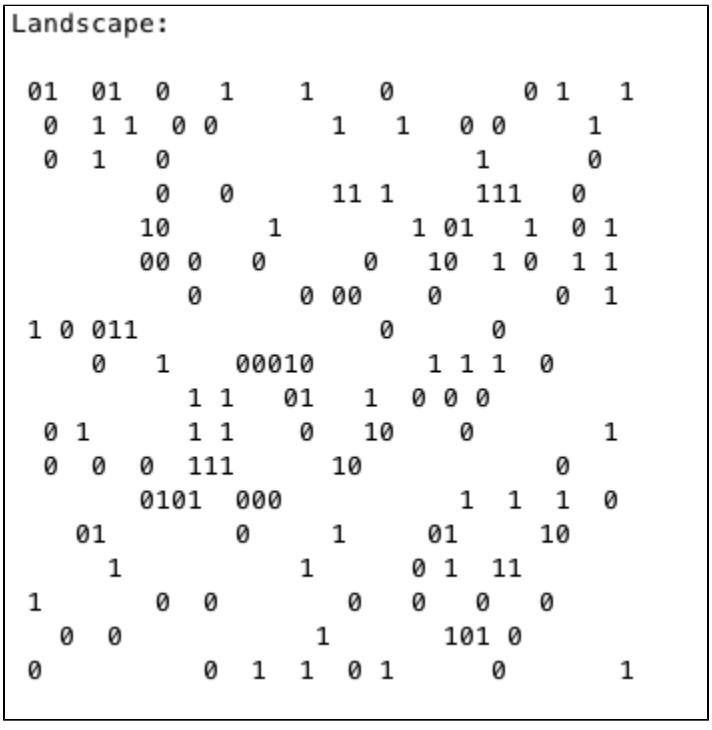
method first creates a 2D array of empty strings, converts landscape to an array list, update each element in the 2D array when it's within the boundary, return a concatenated string. The getNeighbors method returns a list of neighbor Cells within radius 2. The advance method gets a shuffled list of Cells and updates each Cell's state.

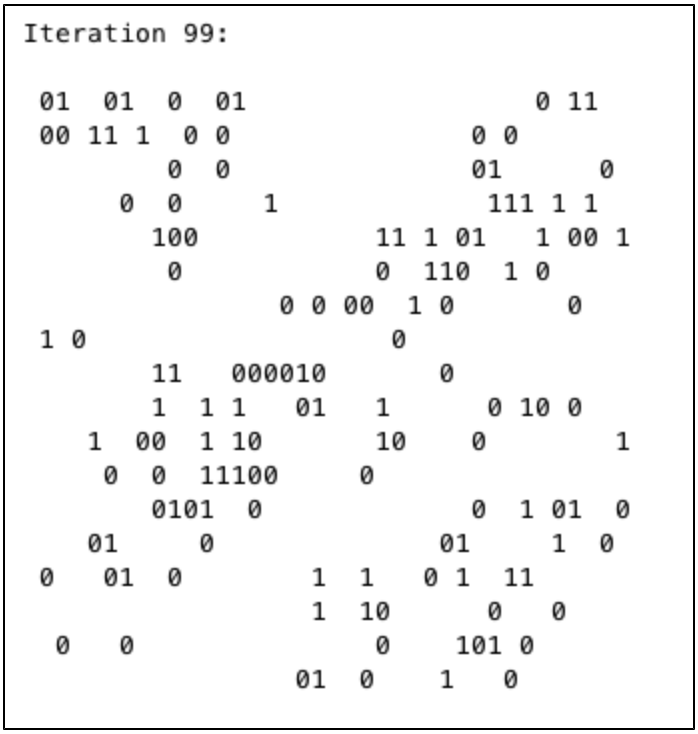
The Cell2 class extends the Cell class and has a different update rule. The toString method returns a single character string indicating the category. The updateState method models the impact on group behavior when neighbors have different number of categories. If there are more of the same category : move +/- 5 with a 1% chance, else : move +/- 5.





The Cell3 class also extends the Cell class and has a different update rule. It models the behavior of sparse Cells. When a Cell has less than 2 neighbors, it moves +/- 5. When a Cell has less than 4 neighbors, it moves +/- 3. Else do not move.
After 100 iterations, the Cells have a clumping behavior, and Cells with the same category also from local clumps.





The Simulation class runs different types of simulation. My first extension is control the parameters using command line arguments.

args[0] = type of simulation

args[1] = number of iterations

args[2] = how often the simulation gets printed to the command line

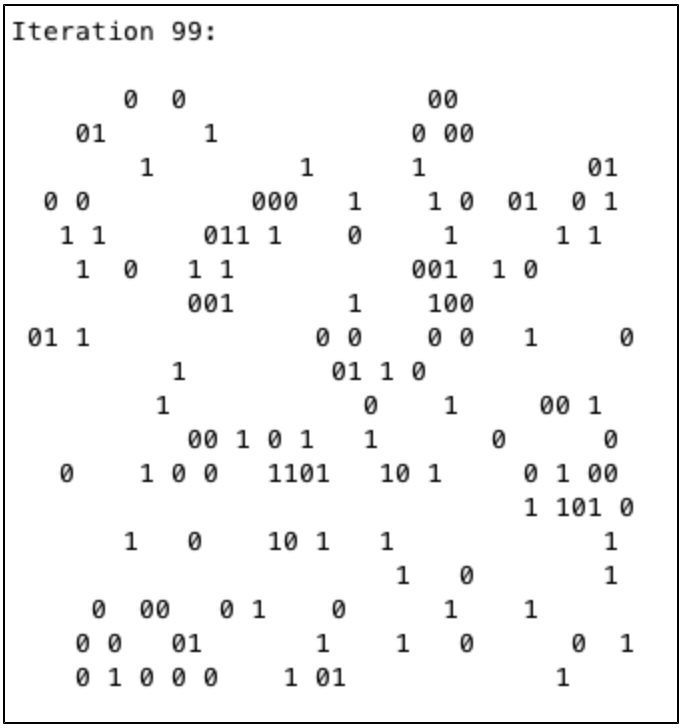
When args[2] == args[1], it prints out all the simulation results. When args[2] <= args[1], it prints out a specified simulation result.

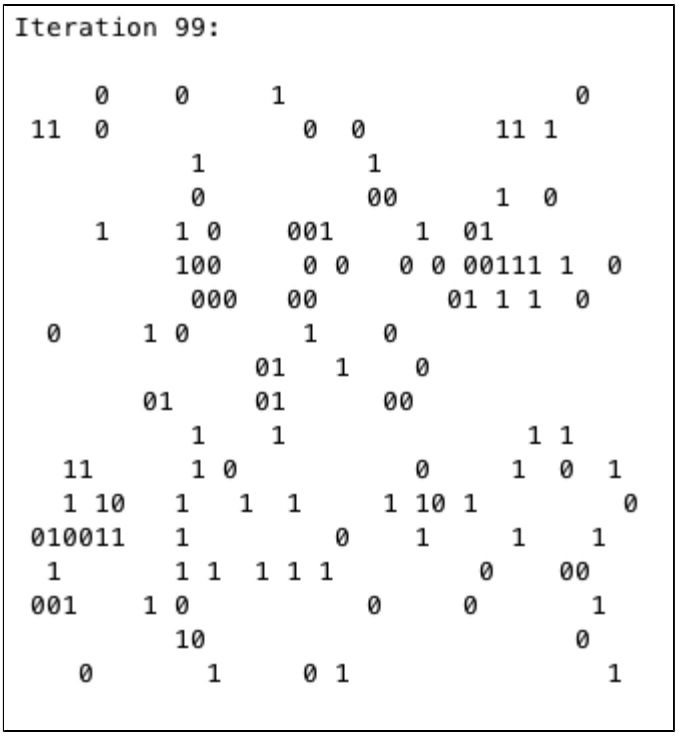
args[3] = width

args[4] = height

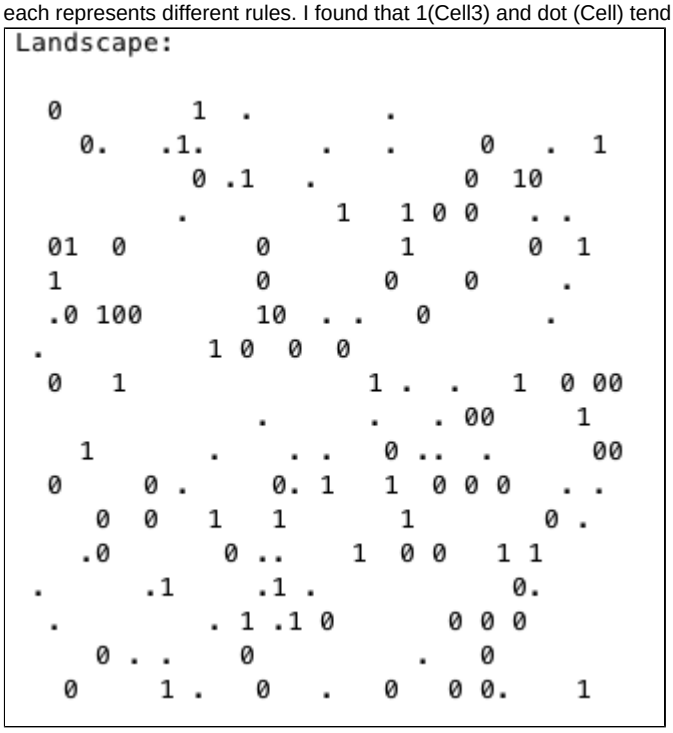
args[5] = number of agents on the Landscape

My second extension is modifying the update rules of Cell3 and analyze the effects. I first changed the rule to move +/- 5 when a Cell has less than 1 neighbor. Then I changed it to move +/- 5 when a Cell has less than 7 neighbor. The second picture showed that increasing the limit of number of neighbors would give longer series of Cells of the same category.

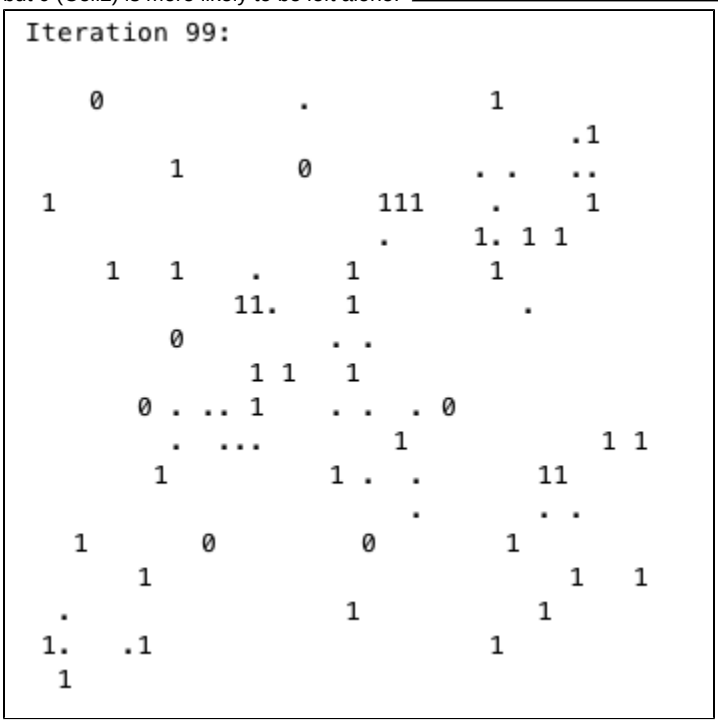

Iteration 99:


Iteration 99:

My third extension is mixing all agents with each represents different rules. I found that 1(Cell3) and dot (Cell) tend to have clumping behaviors,



but 0 (Cell2) is more likely to be left alone.



I got help from Tarini, Akira, and Ian. Tarini was very generous about her time and patience.