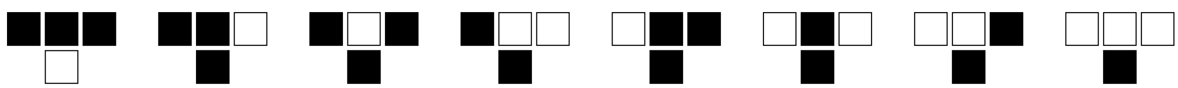
Read: Chapter 4.1: Global dynamics emerging from local interactions: agent-based modeling for the life sciences. By D. Gammack, Elsa. Schaefer, and H. Gaff, pp. 105–113.

1. Find an applet or program that can simulate a cellular automata. One such example is `https://elife-asu.github.io/wss-modules/modules/1-1d-cellular-automata/`. Carry out the following steps for ECA Rules 54, 73, 137, and one more "interesting" rule of your choice. Make the grid size be several hundred; not too big, and not too small (the specific size for which it "looks best" will depend on the specific rule).
   (a) Explore the dynamics of the elementary cellular automaton (ECA) rule using both a single black square as a starting state, and using a random state. Include a screenshot of your findings.
   (b) Which of the four Wolfram classes do you think this rule belongs to, and why?

2. Consider the ECA rule defined by the following *truth table*:



   Here, white = 0, black = 1, and each 4-box diagram represents a three-bit input vector $(x_{i-1}(t), x_i(t), x_{i+1}(t))$ and the corresponding output $x_i(t+1) = f(x_{i-1}(t), x_i(t), x_{i+1}(t))$.
   (a) Find the decimal form of this rule. It should be a number between 0 and 255.
   (b) Write the function $f(x_{i-1}, x_i, x_{i+1})$ as a *polynomial* over $\mathbb{Z}_2 = \{0, 1\}$, by replacing $x \wedge y$ with xy, $x \vee y$ with $x + y + xy$, and $\neg x$ with $1 + x$, and reducing modulo 2.
   (c) Write the local function $f(x_{i-1}, x_i, x_{i+1})$ as a Boolean logical expression (that is, using $\vee$, $\wedge$, and $\neg$).

3. For each of the following, express the ECA rule four different ways: by its truth table, in decimal form, using Boolean logic, and as a polynomial over $\mathbb{Z}_2$.
   (a) the rule that translates its initial state one position to the right;
   (b) the rule that models traffic flow to the left,
   (c) the *logical OR* function: $f(x_{i-1}, x_i, x_{i+1}) = \text{OR}(x_{i-1}, x_i, x_{i+1})$.
   (d) the negation of the logical or function, $f(x_{i-1}, x_i, x_{i+1}) = \text{NOR}(x_{i-1}, x_i, x_{i+1})$.
   (e) the *parity function*, $f(x_{i-1}, x_i, x_{i+1}) = x_{i-1} + x_i + x_{i+1} \pmod 2$.

4. Go to `https://ccl.northwestern.edu/netlogo/` to access NetLogo, which can be run online or downloaded. Find two agent-based models that we did not try out in class. Adjust one of the parameters until you find a "threshold" between two distinct types of behaviors. Describe your findings in a paragraphs, including the approximate value of the threshold, and print out a screen shot (in color, if possible).