Exercises for Math Circles 2018
December 5

1. (a) How many \( k \)-color cellular automata are there?
   (b) How many \( k \)-color totalistic automata are there?
   (c) How many 2D cellular automata are there? (Assume that there are only two colors, black and white.)

\textit{Solution.} (a) There are \( k^3 \) possible color arrangements for a 3-cell neighborhood. For each of these rules, there are \( k \) possible output colors. So there are \( k^{k^3} \) possible rules.

(b) A totalistic rule is defined by the \textit{sum} of the values of the neighbors. Let’s say the possible values are \( 0,1,2,\ldots,k \) (each value represents a color, so there are actually \( k+1 \) colors here if we count 0 as white). Then the possible sums of a 3-cell neighborhood are \( 0,1,2,\ldots,3k \), which counts as \( 3k+1 \) possibilities. And each of these possible states can give \( k+1 \) possible outputs. So the number of \((k+1)\)-color totalistic rules is \((k+1)^{3k+1}\).

(c) If there are are only two states, then there are \( 2^9 \) different possible 9-cell neighborhoods (a cell plus its eight neighbors). Each cell can output two different states. So in total there are \( 2^{2^9} \) possible 2D automata. These are called the \textbf{elementary} 2D automata because there are only two states.

2. Let \( f : \mathbb{N} \to \mathbb{N} \) be the Collatz function:

\[
f(n) = \begin{cases} 
3n + 1 & \text{if } n \text{ is odd} \\
n/2 & \text{if } n \text{ is even.}
\end{cases}
\]

(a) Determine the Collatz sequence starting at 9 — in particular, determine how many iterations are required before the sequence reaches 1.

(b) For a positive integer \( n \), let \( m(n) \) be the least positive integer such that \( f^m(n) = 1 \). (It is an open question whether or not such \( m(n) \) exists!) Find this \( m(n) \) for each of \( n = 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 \).

\textit{Solution.} (a) The sequence is

9, 28, 14, 7, 22, 11, 34, 17, 52, 26, 13, 40, 20, 10, 5, 16, 8, 4, 2, 1, \ldots

You have to iterate the function 19 times to reach 1.

(b) Here is the table:

<table>
<thead>
<tr>
<th>( n )</th>
<th>( m(n) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>7</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td>7</td>
<td>16</td>
</tr>
<tr>
<td>8</td>
<td>3</td>
</tr>
<tr>
<td>9</td>
<td>19</td>
</tr>
<tr>
<td>10</td>
<td>6</td>
</tr>
</tbody>
</table>

3. Let \( g : \mathbb{R}^2 \to \mathbb{R}^2 \) be the function

\[
g(a, b) = (a - b, a).
\]

(a) Describe the orbit of the point \( p = (1, 1) \) under this function.

(b) Describe the set \( \{ n : g^n(p) = (1, 0) \} \).
Solution. (a) You should get the following orbit:

\[(1,1),\ (0,1),\ (-1,0),\ (-1,-1),\ (0,-1),\ (1,0),\ (1,1),\ \ldots\]

After six iterations, we got back to the point \((1,1)\) — this just means \(g^6(p) = p\). So the orbit is periodic.

(b) From (a), we can see that \(g^5(p) = (1,0)\). Since the orbit is periodic, the point \((1,0)\) will occur along this orbit every 6 iterations. So

\[\{n : g^n(p) = (1,0)\} = \{5, 11, 17, 23, 29, 35, 41, \ldots\} = \{5 + 6k : k \in \mathbb{N}\}.\]

4. Rule 90: In this question we will write Rule 90 as a formal cellular automaton.

(a) Write the transition rules for Rule 90.
(b) Find the memory rule \(\mu\) for Rule 90.
(c) Let \(\tau : \{0,1\}^\mathbb{Z} \to \{0,1\}^\mathbb{Z}\) be Rule 90. Define \(\tau\) in terms of \(\mu\). (Specifically, this means find an expression for \(\tau(x)_n\) in terms of \(\mu\) and \(x\), where \(x\) is some state.)

Solution. (a) We’ve done this before actually: 90 in binary is 01011010, so its rule set is

(b) Rule 90 is the XOR automaton, so its memory rule is

\[\mu(a, b, c) := a \ xor \ c.\]

Another way to write XOR is \((x \ xor \ y) = x + y \pmod{2}\).

(c) This is actually the same for any automaton:

\[\tau(x)_n = \mu(x_{n-1}, x_n, x_{n+1}).\]

5. Rule 30: In this question we will write Rule 30 as a formal cellular automaton.

(a) Write the transition rules for Rule 30.
(b) Find the memory rule \(\mu\) for Rule 30.
(c) Let \(\tau : \{0,1\}^\mathbb{Z} \to \{0,1\}^\mathbb{Z}\) be Rule 30. Define \(\tau\) in terms of \(\mu\). (Specifically, this means find an expression for \(\tau(x)_n\) in terms of \(\mu\) and \(x\), where \(x\) is some state.)

Solution. (a) We’ve done this before actually: 30 in binary is 00011110, so its rule set is

(b) It’s tricky to find it, but you might arrive at the following answer:

\[\mu(a, b, c) = a \ xor (b \ or \ c).\]

(c) This is the same as Problem 4! This is how every 1D cellular automaton is defined.

\[\tau(x)_n = \mu(x_{n-1}, x_n, x_{n+1}).\]
6. Let \( \tau : \{0, 1\}^Z \rightarrow \{0, 1\}^Z \) be Rule 126, and let \( x : Z \rightarrow \{0, 1\} \) be the state given as follows:

\[
x_n = \begin{cases} 
1 & \text{if } n = 0 \\
0 & \text{otherwise.}
\end{cases}
\]

(a) Find \( \tau(x)_{-2} \).
(b) Find \( \tau(x)_{-1} \).
(c) Find \( \tau(x)_0 \).
(d) Find \( \tau(x)_1 \).
(e) Find \( \tau(x)_2 \).

**Solution.** The memory rule for Rule 126 is

\[
\mu(a, b, c) = \begin{cases} 
0 & \text{if } a + b + c \in \{0, 3\} \\
1 & \text{otherwise.}
\end{cases}
\]

Now we can use the equation \( \tau(x)_n = \mu(x_{n-1}, x_n, x_{n+1}) \).

(a) \( \tau(x)_{-2} = \mu(x_{-3}, x_{-2}, x_{-1}) = \mu(0, 0, 0) = 0 \).
(b) \( \tau(x)_{-1} = \mu(x_{-2}, x_{-1}, x_0) = \mu(0, 0, 1) = 1 \).
(c) \( \tau(x)_0 = \mu(x_{-1}, x_0, x_1) = \mu(0, 1, 0) = 1 \).
(d) \( \tau(x)_1 = \mu(x_0, x_1, x_2) = \mu(1, 0, 0) = 1 \).
(e) \( \tau(x)_2 = \mu(x_1, x_2, x_3) = \mu(0, 0, 0) = 0 \).
7. **Conway’s Game Of Life.** For each of the following states, use the rules of Conway’s Game Of Life to find (I) the next state, and (II) a predecessor state.

![State 1](image1.png) ![State 2](image2.png)

**Solution.** For the top-left state: here is the next state (left) and a predecessor (right).

![Next State](image3.png) ![Predecessor](image4.png)

For the top-right state, here is the next state. I actually couldn’t find a predecessor — if you find one, let me know!
The next state

For the bottom-left state, here is the next state (left) and a predecessor (right). Note that they are the same!

Finally, for the bottom-right state, here is the next state (left) and a predecessor (right). Note that they are the same as the state itself — it’s a state that never changes!
8. **Conway’s Game Of Life.** Find a state — other than any of the ones in Problem 7 — that never changes.

*Solution.* Here are two:

![Cell diagrams of two stable states in Conway's Game of Life.](image)