**Instruction:** Answer each of the following questions, showing and explaining your work as you go. Partial credit will be awarded based on how well I can follow your work and how far you get, so please use sentences, description, diagrams, and clear definitions to communicate your results as best you can. All diagrams and plots should be labelled, for example.

1. The “neighborhood” in a cellular automata model defines the geometry of the space. In class, we discussed 4 different kinds of neighborhoods for 2-dimensional cellular automata -- 8-neighbor Moore, the 6-neighbor, the 4-neighbor von Neumann, and the 3-neighbor.

   (a) Let $N_k(x, y)$ be the set of neighbors of site $(x, y)$ where $x$ and $y$ are integers. If $k = 3$, then

   $$ N_3(x, y) = \begin{cases} 
   \{(x + 1, y), (x, y + 1), (x, y - 1)\} & \text{if } x + y \text{ is even}, \\
   \{(x - 1, y), (x, y + 1), (x, y - 1)\} & \text{if } x + y \text{ is odd}. 
   \end{cases} $$

   Find similar formulas for $N_4(x, y)$, $N_6(x, y)$, and $N_8(x, y)$.

   (b) In a regular lattice, the atomic loop length is the smallest number of neighboring edges in a loop from $(0, 0)$ back to itself where the same edge between two neighbors is never traversed more than once. Each such loop with this minimal loop length is called an “atomic loop”. For each $k \in \{3, 4, 6, 8\}$, find the atomic loop length and the number of atomic loops containing $(0, 0)$ for lattices with neighborhoods $N_k(x, y)$.

   (c) Given a neighborhood $N_k(x, y)$, we can define a metric $d_k((x, y), (u, v))$ to measure the distance between points $(x, y)$ and $(u, v)$ recursively as follows:

   $$ d_k((x, y), (u, v)) = \begin{cases} 
   0 & \text{if } (x, y) = (u, v), \\
   1 + \min\{d_k((w, z), (u, v)) : (w, z) \in N_k((x, y))\} & \text{otherwise.} 
   \end{cases} $$

   For each $k \in \{3, 4, 6, 8\}$, draw the set of points $\{(u, v) : d_k((0, 0), (u, v)) \leq 2\}$ on the appropriate lattice.

   (d) For each $k \in \{3, 4, 6, 8\}$, find $d_k((0, 0), (3, 3))$.

2. In class, we were able to visually calculate percolation depth because of our awesome pattern analysis wet-ware. Calculating percolation depth algorithmically is a little more complicated, but can be done recursively. See [here](#).

   (a) Let $A(p, N)$ be a random 0/1 matrix with shape $N \times N$ where sites are filled (1) with probability $p$, and empty (0) with probability $1 - p$. Recall that in class, we generated example matrices like this with the python code

   $$ A = \text{floor}(rand(N,N) + p) $$

   For each value of $p \in \{0.3, 0.4, 0.45, 0.5, 0.6\}$, simulate 1000 40 \times 40 matrices. Let $g(x, p)$ be the fraction of these matrices with percolation depth less than or equal to $x$. Plot $g(x, p)$ for each value of $p$, all in one plot. Remember to label your plot axes and include a legend.

   (b) What particular feature of your plot change when $p$ is between 0.45 and 0.5? What does this change mean?

   (c) Suppose we define $h(p, N)$ as the fraction of $N \times N A(p, N)$ matrices which percolate all the way through. Plot $h(p, N)$ as a function of $p \in [0.3, 0.6]$ (this is an INTERVAL, not a set!) for $N \in \{5, 10, 20, 50, 100\}$. Use at least 1000 matrices for each. (Warning: This calculation may take you a long time.)
(d) Extrapolating from your plot, what do you think will happen to \( \lim_{N \to \infty} h(p, N) \)?

(e) Reality, of course, is often three-dimensional, rather than two-dimensional. Describe how you think the percolation threshold will change when we switch from two dimensions to three dimensions and why it will change.

3. In Conway’s game of life, suppose you start with a straight line of 7 live cells. Draw the first 3 steps, and the equilibrium configuration.

4. The pendulum equation can also be studied in the phase-plane.

   (a) Rewrite the dimensionless pendulum equation \( \ddot{\theta} = -\sin \theta \) as a system of two first-order equations.

   (b) Modify this code to generate a phase-plane stream plot of your system of two first-order equations. Your domain should be large enough to include exactly 5 steady-states.

5. As nuclear physics predicted and history has shown, standard nuclear power plants have an inherent vulnerability. Under stable operation, the heat a reactor generates is dissipated by it’s enveloping cooling system as quickly as it is generated. However, as the reactor temperature increases, the fission rate increases. There is a critical tipping point at which heat is generated faster than the cooling system can compensate, fission becomes a run-away chain reaction, and we get a meltdown. A simple model of this phenomena can be made by adding a nonlinear growth term to the equation for temperature. Let \( \theta(x, t) \) represent the reactor’s temperature over a one-dimensional cross-section from \(-L/2\) to \(L/2\). At the boundaries, the temperature \( B \) is set by the cooling system. Within the reactor, the temperature \( \theta \) obeys the non-dimensionalized partial differential equation

\[
\frac{d\theta}{dt} = \frac{d^2\theta}{dx^2} + e^\theta.
\]

   (a) What is the differential equation the temperature distribution will solve at steady-state?

   (b) Where, between \(-L/2\) and \(L/2\) does the reactor reach its maximum temperature?

   (c) Rewrite your steady-state equation as a first-order system.

   (d) Numerically solve your first-order steady-state system for \( x \in [0, 2.5] \) for maximum temperatures \( m \in \{-1, 0, 1, 2, 3, 4\} \), and plot your solutions.

   (e) If the cooling system keeps the boundary temperature \( B = 1 \), what is the largest reactor size \( L^* \) for which a steady-state temperature profile exists.

   (f) If \( L = 3 \), what is the cooling system’s maximum allowed temperature \( B^*(L) \) for which a steady-state temperature distribution can exist?

   (g) How do you interpret the fact that for \( B < B^*(L) \), there are two different steady-state solutions satisfying the boundary conditions?

6. William Thomson’s flat-earth heat equation solution

\[
\theta(x, t) = \theta_0 \text{erf} \left( \frac{x}{2\sqrt{\kappa t}} \right),
\]

can only be a good approximation for the earth if the changes in temperature are confined to the upper surface of the earth. Using scipy.special.erf, plot this solution at time \( t = 200,000 \) from \( x = 0 \) to \( x = 7 \times 10^6 \) when \( \kappa = 2.8 \times 10^3 \). Is the flat-earth approximation reliable?