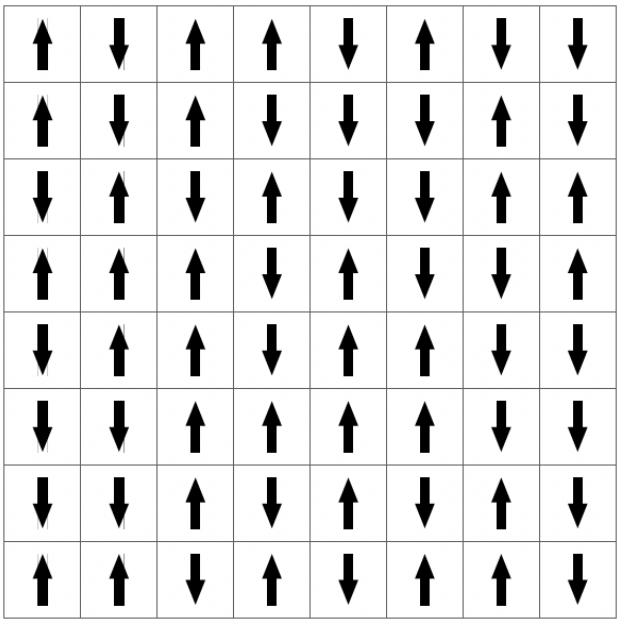
Assignment 9A

The Ising model is a fundamental model in statistical mechanics that describes ferromagnetism using discrete spins arranged on a lattice. Each spin $\sigma_i = \pm 1$ interacts with its nearest neighbors, and the system's energy is given by:

$$E = -J \sum_{\langle ij \rangle} \sigma_i \sigma_j \,, \tag{1}$$

where $J > 0$ corresponds to a ferromagnetic interaction.



As explained in class, at low temperatures, the spins tend to align, leading to spontaneous magnetization. As the temperature increases, thermal fluctuations compete with the interaction energy, eventually destroying long-range order and driving the system into a paramagnetic phase.

In this assignment, we will simulate the Ising model using the Metropolis algorithm, as discussed in Lecture 9.

- **Problem 1 (100 points):** The Ising Model

  Write a program to perform a Markov Chain Monte Carlo (MCMC) simulation of the Ising model on a 20×20 square lattice. You will need to set up variables to hold the value ± 1 of the spin on each lattice site, probably using a two-dimensional integer array, and then take the following steps.

  (a) First write a function to calculate the total energy of the system, as given by the equation above. That is, for a given array of spin values, go through every pair of adjacent spins and add up the contributions $\sigma_i \sigma_j$ from all of them, then multiply by $-J$.

  Hint: Each unique pair of adjacent spins appears only once in the sum. Thus, there is a term $-J\sigma_1\sigma_2$ if spins 1 and 2 are adjacent to each other, but you do not also need a term $-J\sigma_2\sigma_1$.

  (b) Implement a Metropolis-style simulation of the Ising model with $J = 1$ and temperature $T = 1$ in units where the Boltzmann constant k_B is also 1.

  As explained in Lecture 9, start by initializing the lattice with random spins (e.g., sampling them independently as ± 1 with equal probability would result in a total initial magnetization close to zero, as on average half of the spins are up and half are down). At each step of the simulation:

  * Select a spin at random.
  * Calculate the energy difference ΔE that would result from the flipping of the chosen spin.
  * Apply the Metropolis acceptance criterion to determine whether to accept the flip.
  * If the move is accepted, update the spin; otherwise, the spin remains unchanged. In both cases, the configuration counts as a step in the Markov Chain.

  Repeat this process for a sufficiently large number of Monte Carlo steps. A reasonable choice is on the order of 10^6 steps, as used in the example code.

  (c) Make a plot of the total magnetization

  $$M = \sum_i \sigma_i \tag{2}$$

  as a function of the number of Metropolis algorithm steps. If everything is done correctly, you should see that during the initial steps the magnetization fluctuates unpredictably before settling into equilibrium fluctuations around a well-defined value.

  (d) Run your program again, but this time set the temperature to $T = 2.5$, where the system is expected to be in the paramagnetic phase.