Overview: Homework is due a week from Monday March 26 before class.

The Ising model of magnetism is defined by the following relation which returns an energy E for a configuration of spins S_i,

$$E = J \sum_{\langle ij \rangle} S_i S_j \tag{1}$$

The sum is taken over nearest neighbors bonds of an $N_s = L \times L$ square lattice with periodic boundary conditions (PBC). Note that the number of bonds is $2N_s$ for square lattice with PBC. $S = \pm 1$ and since we are interested only in ferromagnetic interactions we will set $J = -1$ for the rest of this homework.

We will need essentially only one important result from statistical physics for this project, the formula for the thermal average of an observable O.

$$\langle O \rangle = \frac{\sum_c O e^{-\beta E}}{Z} \tag{2}$$

where $\beta = 1/T$ is the inverse temperature. $e^{-\beta E}$ is called the weight of a configuration (it is also often referred to as the Boltzmann factor), $Z = \sum_c e^{-\beta E}$ is the normalization, called the partition function (cf: `https://en.wikipedia.org/wiki/Boltzmann_distribution`; we have chosen to work in units so that $k_B = 1$). The sum on c runs over all 2^{N_s} configurations of the Ising spins. Examples of observables could be E itself or the magnetization $M \equiv \sum_i S_i$; we will see more examples below.

I. OBSERVABLES

We will now rewrite two important thermodynamic quantities that are experimentally accesible, the specific heat and susceptibility directly in terms of thermal averages. Prove analytically,

$$C_v \equiv \frac{1}{N_s}\frac{d\langle E \rangle}{dT} = \frac{1}{N_s T^2}\left(\langle E^2 \rangle - \langle E \rangle^2\right) \tag{3}$$

$$\equiv \frac{d\langle e \rangle}{dT} = \frac{N_s}{T^2}\left(\langle e^2 \rangle - \langle e \rangle^2\right) \tag{4}$$

where the second line has been defined using intensive quantities, *i.e.* $e = E/N_s$.

Similarly prove that,

$$\chi \equiv \frac{1}{N_s}\left.\frac{d\langle M \rangle}{dh}\right|_{h=0} = \frac{1}{N_s T}\left(\langle M^2 \rangle - \langle M \rangle^2\right) \tag{5}$$

$$\equiv \left.\frac{d\langle m \rangle}{dh}\right|_{h=0} = \frac{N_s}{T}\left(\langle m^2 \rangle - \langle m \rangle^2\right) \tag{6}$$

where $E \to E - hM$ in the presence of a magnetic field, h. $M \equiv \sum_i S_i$ and $m = M/N_s$. These formulae allow you to conveniently calculate χ and C_v directly from thermal averages without evaluating any derivatives. *(20 points)*

II. FULL ENUMERATION

For very small lattices we can evaluate the thermal averages directly by enumerating all the 2^{N_s} terms in the sum Eq. 2. We are interested in the averages $\langle e \rangle, \langle e^2 \rangle$ and $\langle m^2 \rangle$ (so we can compute the specific heat and the susceptibility; note that without an external field $\langle m \rangle = 0$ by symmetry). Your main exercise this week is to write a python program "ising_enum.py" that evaluates Eq. 2 by enumerating all the 2^{N_s} terms that appear in the thermal averages. Carry this out in the following steps:

(1) Run and understand the program "ising_enum_hw.py" given to you. It shows you how to enumerate all states in the Ising square lattice. It also computes and prints out the magnetization of every state correctly. In its current form it only prints "0" for the energies. Complete the function "get_ener.py" so that it computes the energy of any $L \times L$ spin configuration according to Eq. 1 with PBC and $J = -1$. When the program is run it should now correctly print out the magnetization and energy. For a 2×2 system with PBC note that each bond is counted twice so e.g. for all spins pointing up $E = -8$. *Hint:* There are many ways to compute the energy. The numpy function "roll" make this very easy and avoids nested for loops which are slow in python. If you run and understand the program "roll.py" it contains all the ideas you will need to complete this homework, but you are encouraged to research the function yourself.

(2) Now that we can enumerate all the Ising states and compute their magnetization and energy, we can easily evaulate the sums of the type Eq. 2 to get $\langle e \rangle, \langle e^2 \rangle$ and $\langle m^2 \rangle$. Complete and debug this code, which we shall call "ising_enum.py".

(3) With you new program, make a plot of the $E(T)$, $C_v(T)$ and $\chi(T)$ computed for a range of T and for a few values of L you can run reasonably. What do you observe?

Note that full enumeration is exponentially hard. The computational effort grows exponentially with L, so it is not very practical. We will use this full enumeration code as a test for more sophisticated algorithms that we will develop in the following weeks. *(80 points)*