Tutorial 4 Tasks:

1. Using your `ising_mc.py` code from Tutorial 1, generate Monte Carlo (MC) training samples at various temperatures. Then read them into line 47 of `train_ising2d.py`. Start with training data on an LxL lattice, with `L=4`.

2. For several different temperatures, run '`python train_ising2d.py`' to train RBMs (one machine for each temperature) with 4 hidden neurons. This code will save the parameters (weights and biases) to files within the directory `/data_ising2d/RBM_parameters/`.

3. Run '`python sample_ising2d.py`' (modify line 48 to read in your own trained parameters) to generate new samples. This code will compute physical observables <E>, <C>, <M>, <chi> on your visible samples and save them to files within the directory `/data_ising2d/RBM_observables/`.

4. Run '`python plot_results.py`' to plot your samples' expectation values and compare with "exact" (Monte Carlo) results (blue). The RBM results for a well-trained machine are plotted in orange.

5. Repeat Tasks #2-#4 for other numbers of hidden units. How many hidden units do you need to converge to 4x4 MC results? What about 6x6 or 8x8 MC results?