0.4 General pre-prep log

15/01/2026

Added recommended textbook to bibliography [**?**].

This lab explores the dynamics of the ising model; a simple model magnet

19/01/2026 Briefly discussed lab w/Dr Turci; he noted the "percolation" chapter in [**?**] as a good starting point since it introduces many ideas relating to phase transitions that will be useful for this project

Blackboard resources [**?**] "ising1.pdf" have provided a general overview of background concepts and information. Useful ideas and concepts summarised in "0.5 background theory and concepts." These are NOT the full background theory for this lab, but represent a useful starting point.

21/01/2026 Called lab partner to introduce self; went over basic logistics and plans, are now on same page.

Bb materials "earmark" list:

- "ising1.pdf", "ising2.pdf" - recaps basic stat mech theory, and covers basic setup of ising model
- "ising exercise.ppt" - notes metropolis algorithm, alongside some observables to compute
- "finite-size-effects.pdf" - goes over phase transition details - order parameters and critical exponents -

Looking through more of the BB materials; and trying to think of vague plan:

- Go through BB "finite size effects.pdf"
- Go through "percolation" chapter [**?**] - (in retrospect this may take a while) - note section 1.8 supposedly outlines extraction of critical exponents
- Familiarize self w/python simulation provided
- Refactor code in a .py file - jupyter notebooks are more annoying to work with in my experience
  - Additional benefit: lets me use my personal library [**?**] which may be useful
- Refactor/expand code "architecture" generally - some potential ideas:
  - MAYBE rework simulation to allow arbitrary dimensionality (I.E, 1D, 2D, 3D, 4D…) - suspect there may be interesting comparisons for different dimensionalities
  - Rework as object orriented - each config could be it's own object with observable methods (I.E, computeEnergy, magnetisation…) and simulation step method - a "simulation run" class could then act as a wrapper for many config objects over time

        * May also need to handle different simulations “in parallel” - I.E, need to vary sim parameter to generate data points
        * Maybe set it up s.t configuration is object w/update method; overwrites itself; is a data destructive idea, however should only need a final “equilibrium” configuration for a given set of sim parameters
            · Then a “dataset” object would be a list of final configs + some metadata (I.E, lattice size, # sim steps, dimensionality…)
    - Export/import method - given the computational nature of this lab, heavy compute runs may occur - being able to save raw simulation runs data in a standard format may save compute time
    - Additional observable computation - current code computes energy and magnetisation - adding additional observables may be useful for investigation, I.E:
        * Heat capacity
        * Susceptibility
        * Helmholtz free energy
- Maybe setup shared github repo w/lab partner