# Computer Simulation I – Computational Methods Assignment 3

The purpose of this assignment is to explore:
- the physics of the 2D Ising model
- the use of generative AI as a tool for computer simulation

## Tasks

1. Using Microsoft Copilot, generate a Python class to simulate the 2D Ising model on a square lattice using the Metropolis Monte Carlo algorithm.
2. Test and critically evaluate the response given by Copilot, and make (at least) 3 modifications/ improvements to your new Python class.
3. Plot the average magnetisation $\langle M \rangle$, energy $\langle E \rangle$, magnetic susceptibility $\chi$, and heat capacity $C$ as a function of the magnetic field $h$ for system size $L \times L = 10$ and two different temperatures: $k_B T/J = 1.0$ and $k_B T/J = 4.0$.
4. Plot the same quantities as a function of temperature for $h = 0$. Find the critical temperature $T_c$ of the phase transition between ferromagnetic and paramagnetic phases.
5. Plot the magnetic susceptibility and the heat capacity as a function of system size at $T = T_c$ in order to extract the critical exponents.
6. *Bonus:* Explore the physics of the model using your code; in particular, you may wish to consider the effect of:
   - system size and shape
   - initial state
   - number of Metropolis samples
   - sweeping through lattice sites sequentially or randomly
   - adding an external field
   - spatial dimensions greater than 2
   - other lattice geometries (e.g. hexagonal, triangular, etc.)
   - different boundary conditions
   - methods to visualise the microstate
   - ….feel free to explore and get creative!

## Report

The assignment is assessed through a written report, in the same style as an experimental lab report. The "experiment" here is twofold: you are experimenting with the AI, and you are also experimenting with the 2D Ising model.
The report should comprise a maximum of **6 pages (A4)** and contain the standard sections:
- **Introduction** – this should explain the
- **Methods** – this should include a description of your prompt(s) to the AI, its response(s), and how you improved on the code in order to solve the assignment
- **Results** – this should include the plots specified above together with a discussion of their physical meaning
- **Conclusions** – this should include a critical assessment of the AI's capability as a tool to assist with computer simulation

Your report *should* include:
- A general introduction to the Ising model & generative AI, and why they are interesting/ relevant *(in the Introduction section).*
- Your initial AI prompt(s) and a description of its response(s) *(in the Methods section)*. You may include short snippets of code in the report if you wish – you will submit the full code generated initially by the AI in a separate file (see **Submission** below).
- Well presented plots including descriptive figure captions and labelled axes *(in the Results section).* Any claims you make should ideally be substantiated with quantitative results.
- A critical evaluation of the role of the AI in the process – where was it useful, and where did its output require human improvement/intervention? *(in the Conclusions).*

Your report *should not* include:

- Any AI-generated content without explicitly citing/acknowledging the source. **Submitting AI-generated content without acknowledgment is plagiarism and will have very serious consequences.** Any AI-generated content must be reported by quoting the source (in this case, Microsoft Copilot), the prompt used to generate it**,** and the date and time at which it was generated.
- Long sections of code – use short snippets if you want to specifically illustrate something within the report, otherwise refer to your code that you will also submit (see **Submission** below).
- A detailed explanation of the Metropolis Monte Carlo method or large-language models – you may assume that these are already familiar to the reader.

## Submission

Your submission should include (at least) 3 files:

- Your report **in PDF format**.
- The code initially generated by the AI (e.g. as a .py or .ipynb file).
- Your final code that was used to generate the results and plots. **This should be one or more executable Python files (.py or .ipynb)** that, when run, generate all plots shown in the report.