

MAS2008 Scientific Computing – Lecture 1

Neil Strickland

February 1, 2024

What is this course about?

- ▶ **Aim:** by the end of the course, you should be qualified to contribute meaningfully to a scientific computing project.
- ▶ You should be able to write Python code for various tasks.
- ▶ You should have understood a selection of algorithms.
- ▶ You should be able to use the available tools:
 - ▶ The VS Code integrated development environment
 - ▶ Debugging systems
 - ▶ AI services for code generation
 - ▶ Git for version control and collaborative coding (baby steps only)
 - ▶ Systematic unit testing (baby steps only).
- ▶ You should be familiar with the ecosystem:
 - ▶ Standard large packages: `numpy` (numerical calculation), `matplotlib` (plotting), `pandas` (data analysis) etc.
 - ▶ How to find, install and use more specialised packages.
 - ▶ How to find and understand the documentation, or use AI to avoid that.
 - ▶ How to find datasets and public APIs, and interact with them programmatically.

Course activities

- ▶ There will be one lecture every week. Usually I will write code live and talk about it.
- ▶ Often I will also distribute a tidier and more complete version of the code, possibly with associated YouTube videos.
- ▶ There will be one lab session every week.
- ▶ You will be asked to complete various tasks, which will partially overlap with tasks covered in the lecture.
- ▶ For some of these tasks, you will be asked to paste code into an online test system.
- ▶ There will be three assignments, which will account for most of your grade.
- ▶ For each assignment, you will be asked to submit a Jupyter notebook containing a mixture of Python code and text with mathematics written in LaTeX.
- ▶ Most of the credit will be for intelligent and accurate analysis of relevant mathematical issues and how they interact with software issues. Only a smaller part of the credit will be for the code itself.
- ▶ Assignments will be extended versions of tasks introduced in the lab sessions.

AI assistance

- ▶ The situation with AI coding assistants is new, exciting and rapidly changing.
- ▶ Most professional software engineers now use these tools.
- ▶ You are also encouraged to use them. However:
 - ▶ You must not submit code that you do not understand.
 - ▶ If an AI gives you code that you do not understand, you can ask the AI to explain it, search the documentation, modify and test the code, and continue until you do understand it.
 - ▶ In the assignments you must explicitly acknowledge any AI assistance that you used. The assignments will contain more detailed instructions for this.
- ▶ The AI picture will certainly change greatly in the coming years. Specific knowledge from 2024 will expire, but you should aim to develop skills that will help you to keep up.
- ▶ I hope and expect that students will discover and tell me about interesting things that I do not already know.

- ▶ You are also encouraged to talk to other students. However:
 - ▶ You must not copy code or text directly from other people. You can develop your understanding by talking to them, but then you must go away and write your own code and text based on that understanding.
 - ▶ You must not submit code that you do not understand.
 - ▶ In the assignments you must explicitly acknowledge any other people with whom you have discussed the problems.
- ▶ You are free to search the internet for information (and some tasks will actually tell you to do that). However:
 - ▶ You must not submit code that you do not understand.
 - ▶ In the assignments you must explicitly acknowledge any internet sources that you have used.