

Computer Simulation I

PYU33C01

Computational methods in Python

Course overview

1. Numerical methods – Prof. Stefan Hutzler

- How to convert a physics problem into an algorithm that can be run on a computer
- Largely independent of programming language

2. **Computational methods – Dr. Mark Mitchison (me)**

- How to implement algorithms, analyse data, and visualise results
- Focus on Python language
- Exploit existing libraries to solve complex problems efficiently

Course structure

- Lectures:
 - introduce concepts and demonstrate code
- Labs:
 - put what you have learned into practice
- 3 assignments, equally weighted
- Assignments 1 & 2:
 - solve computational problems inspired by the labs
 - submit code electronically via Blackboard
- Assignment 3:
 - experimentally assess the capabilities of AI as a coding tool
 - write up findings in a lab report (submit electronically via Blackboard)

100% continuous assessment = no exam!

Syllabus

Lectures:

- Python basics

Lab 1 → Assignment 1

- Scientific computing with NumPy and SciPy

Lab 2 → Assignment 2

- Object-oriented programming
- Generative AI as a coding tool

Lab 3 → Assignment 3

Getting set up

Course requirements:

- Python 3
- Scientific libraries
- Notebook environment
- All provided by **Anaconda distribution**
- Available in computer lab but installation on home computer/laptop is **highly recommended**
- Download and install at <https://www.anaconda.com/download/>

