Knowledge Exchange Hub *for Mathematical Sciences*

UK KE Hub Mathematics

Session 1 — Introduction to AI-Assisted Coding

Session 1 — Introduction to AI-Assisted Coding

Duration: 60 minutes

- Welcome & course overview (5m)
- What AI-assisted coding is (10m)
- Tools & environments (10m)
- Responsible use & IP considerations (15m)
- Short hands-on warmup (15m)

Welcome & Course Overview

- Course title: Coding and Programming Using AI for Efficient Coding
- Sessions: 5 short sessions (concepts + hands-on)
- Delivery: Microsoft Teams + course lab repo
- Hands-on: exercises in code-with-ai-lab (zip available if needed)

Learning objectives

By the end of this course participants will be able to:

- Understand how AI assistants (Copilot, ChatGPT, etc.) fit into a developer workflow.
- Use AI suggestions safely and evaluate generated code.
- Apply basic IP/licensing checks and document AI usage.
- Integrate AI into testing, debugging and refactoring workflows.

What is AI-assisted coding?

- AI-assisted coding = tools that suggest code (snippets, completions, tests, docs) based on context.
- Examples: GitHub Copilot, Copilot Chat, ChatGPT (code models), TabNine, Codeium, AWS CodeWhisperer, Google Collab.
- Modes: inline code completion, conversational chat, batch generation, documentation helpers.

Why this matters

- Productivity: faster prototypes, boilerplate generation.
- Risks: hallucinations, license/usage obligations, over-reliance, data leakage.
- Responsibility: review, test, document.

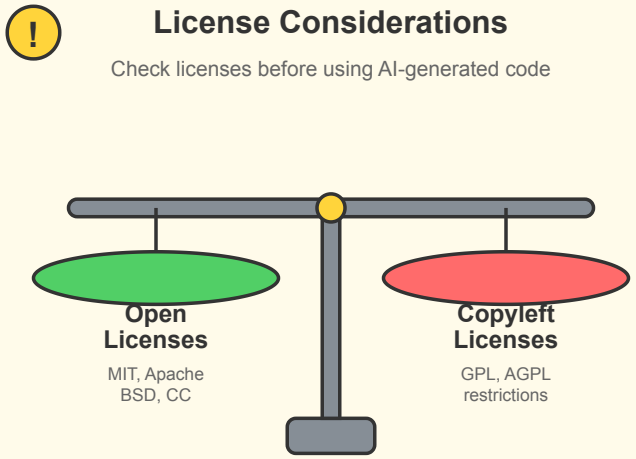
Tools & environment (what we'll use)

- VS Code (recommended) — https://code.visualstudio.com/
- GitHub for repositories (students may use clones or ZIPs) — https://github.com/
- GitHub Copilot (optional) — https://github.com/features/copilot
- Other tools: ChatGPT, TabNine, Codeium (for students without Copilot)

Responsible use — quick checklist

- Never paste proprietary secrets / credentials into public AI tools.
- Treat generated code as a suggestion: read, test, and adapt.
- Check for matching code and license obligations for non-trivial code.
- Document AI assistance in comments and README.

IP & Licensing (concise)



- If generated code matches external code, the original license applies
- **MIT/Apache 2.0** — permissive (OK with attribution) • **GPL/AGPL** — copyleft (may require same license)
- For institutions: follow internal IP policy and consult legal advisors

Demo: quick example (live)

1. Open VS Code and a small starter file.
2. Ask Copilot or trigger suggestion for a function (e.g., parse a CSV line).
3. Inspect the suggestion, run tests, and fix issues.

Warmup hands-on (15m)

- Goal: make everyone comfortable with opening the lab repo and running a test.
- Steps (see handout):
    i. Clone or unzip the lab repo.
    ii. Open exercise 01 in VS Code.
    iii. Run tests: `pytest -q` (or run using the Python extension).
    iv. Use an AI assistant to suggest an implementation (for example, modify the python script to interact with the user), review, and run tests.

Q&A and wrap-up

- Quick recap of key points.
- What to expect in Session 2: deeper tool setup and hands-on exercises.
- Resources and handouts location (link to repo).