**Week 2 Lesson Plan: Debugging & Iterative Thinking**

**1. Overview:**

This session introduces students to debugging AI-generated scripts, identifying common errors, and developing structured problem-solving strategies. Students will refine AI-generated solutions and compare them with human-optimized approaches.

**2. Learning Goals & Key Principles:**

By the end of this session, students will be able to:

* Identify and correct **common AI-generated coding errors**.
* Apply **structured debugging techniques**.
* Develop an **iterative approach** to improving AI-generated code.
* Compare **multiple AI-generated outputs** to assess variations in AI performance.
* Implement **unit testing** to verify correctness.

Key principles include:

* **Debugging as an Essential Skill** – Understanding why debugging is critical.
* **Iterative Thinking** – Refining and improving AI-generated code through multiple cycles.
* **Testing & Validation** – Using unit tests to confirm correctness and efficiency.

**3. Mini-Lecture: Introduction to Debugging (Step-by-Step)**

**(15 min total)**

1. **Common AI Coding Mistakes & Debugging Strategies** (5 min)
   * Overview of syntax errors, logic errors, and inefficiencies.
   * Case studies of AI-generated coding issues.
2. **Debugging Approaches** (5 min)
   * Step-by-step debugging strategies.
   * Live demonstration of debugging a faulty AI-generated script.
3. **Introduction to Unit Testing** (5 min)
   * Writing simple unit tests in Python/R.
   * How unit tests prevent errors in iterative coding.

**4. Hands-on Debugging Challenges**

**(60 min total, divided into three core tasks)**

**Task 1: Identifying AI Errors & Debugging (20 min)**

* Students review **faulty AI-generated scripts**.
* **Error classification:** Syntax, logic, data handling issues.
* **Students correct errors & discuss debugging approaches.**

**Task 2: Iterative Improvement of AI Code (20 min)**

* Students **run the same prompt multiple times** in ChatGPT.
* **Compare AI-generated variations** to analyze inconsistencies.
* Students refine prompts to **improve AI’s coding accuracy**.

**Task 3: Unit Testing & Validation (20 min)**

* Students write **unit tests** to verify correctness of debugging solutions.
* **Group discussion:** How unit tests ensure reliability.
* **Instructor review of best practices for testing.**

**5. Wrap-Up & Reflection (15 min)**

* **Group Discussion:** Debugging best practices & lessons learned.
* **Key Takeaways:** Importance of structured debugging & testing.
* **Looking Ahead to Week 3:** Scaling complexity with algorithmic thinking.

**Appendix: Additional Exercises**

**Comparing Multiple AI-Generated Solutions**

1. **Prompt Variation Exercise:**
   * Students generate **the same code task using different prompt variations.**
   * Compare how AI **interprets different phrasings.**
   * Discuss **how to refine prompts for better accuracy.**
2. **Code Efficiency Comparison:**
   * Run **two AI-generated scripts** and compare execution time.
   * Identify which script is more optimized and why.

**Unit Testing Exercises**

1. **Writing Unit Tests for Data Processing Functions:**
   * Students write test cases to check for **missing values, incorrect calculations, and invalid data types**.
2. **Automating Testing in an Iterative Workflow:**
   * Students modify AI-generated scripts **incrementally**, rerunning tests to ensure correctness.
   * Discussion: How does testing impact iterative debugging?

**Outcome:**

By the end of this session, students will have experience debugging, improving, and validating AI-generated code through structured testing and iterative refinement.