**Subgroup M**  
M.1 — [S13M1] Stable sort employees by dept asc, salary desc  
Scenario (agritech):  
Context:  
HR exports in agritech require deterministic sorting for payroll audits.  
Your Task:  
Sort employees by dept ascending and salary descending (stable), and re-emit CSV.  
Data & Edge Cases:  
name,dept,salary rows provided.  
AI Assistance Expectation:  
Use AI to outline csv.DictReader/Writer usage and key composition.  
Constraints & Notes:

Stable sort within department by salary desc.  
**Sample Inpu**t  
name,dept,salary  
Raj,Eng,120  
Maya,HR,90  
Abi,Eng,110  
**Sample Output**  
Raj,Eng,120  
Abi,Eng,110  
Maya,HR,90  
Acceptance Criteria: Stable and correct ordering

**Prompt:**

**Reads employee data from input.csv sorts by department(a-z) and salary(high to low) print each row with field separated by comma and space,and add the comments**

**Code: A screen shot of a computer code

AI-generated content may be incorrect.**

**Output:**

A computer screen shot of a computer program

AI-generated content may be incorrect.  
M.2 — [S13M2] Validate Palindrome Function with Tests  
Scenario (AI-assisted debugging):  
In an agritech text-analysis module, engineers need a function to check whether a given  
string is a palindrome (ignoring case and non-alphanumeric characters). A previous intern  
wrote a buggy version of the function.  
Your Task:  
1. Use AI-assisted coding tools (e.g., VSCode Copilot, Gemini, CursorAI) to:  
o Debug and correct the provided function.  
o Write at least 5 unit tests (positive, negative, edge cases).  
2. Ensure the function passes all your test cases.  
Buggy Starter Code (given to students):  
def is\_palindrome(text: str) -> bool:  
cleaned = ''.join(ch for ch in text if ch.isalpha())  
return cleaned == cleaned[::-1]  
Data & Edge Cases:  
• Input may contain spaces, punctuation, and digits.  
• Must be case-insensitive.  
• Empty string should return True.

Deliverables:  
• Corrected Python function.  
• A test suite (you may use pytest or Python’s built-in unittest).  
• Explanation of what bug(s) you fixed.  
**Sample Input / Output**:  
Input: "A man, a plan, a canal: Panama"  
Output: True  
Input: "hello"  
Output: False  
Acceptance Criteria:  
• Handles mixed case + non-alphanumeric characters.  
• Passes at least 5 meaningful test cases (normal, edge, tricky cases).  
• Code must be readable and AI-generated documentation (docstring/comments)  
included

**Code:** A screenshot of a computer program

AI-generated content may be incorrect.

**Output:**

**A screenshot of a computer error

AI-generated content may be incorrect.**