Notes for instructor:

**Grading by TAs:** If TAs will be grading these assignments, specifically checking the answers, we might need to introduce formatting requirements to ensure the output is easily readable. The output might be difficult to interpret, especially if students use the print(VAR) function.

**Output file saving:** Expanding on the previous point, we should consider asking students to save their outputs to files. This would make it straightforward to develop a Python script for grading that could automatically compare the student's output with the expected output from the solution script. To facilitate this, we can provide students with a snippet of code to guide them, such as:

with open(PROBLEM\_FNAME, "w") as f:

f.write(ANSWER\_VAR)

**Library use restrictions:** We might want to limit the third-party libraries students can use to prevent reliance on obscure or complex libraries. A practical approach would be allowing only those packages with the Anaconda installer. Students can find the list of these packages at: https://docs.anaconda.com/anaconda/packages/py3.9\_linux-64/. The last column with checkmarks shows the packages included with the installer.

**Implementation requirements section**: I have added a section on implementation requirements to the write-up. Currently, it specifies two rules: firstly, students must use Python 3 rather than Python 2; and secondly, they must submit their work either as a single Python script or as a Jupyter Notebook.