

ProgSD lab exam general feedback (December 2024)

The lab exam was open book. We marked the code for both parts manually, giving partial credit for minor errors where the intention was clear. Note that if we had used fully automatic marking, many students would have received a zero for not following instructions.

Open-Book Exam issue: Many students relied heavily on online searches instead of using pre-existing code from the practical labs.

We used an online code similarity tool to check for possible collusion, and are happy that we found a low amount of collusion in this year's submissions. Students where there was a possible conduct issue have been contacted by the Student Conduct team.

Python: The Python Lab Exam required solving three tasks in a single Python source file, covering Python basics, database management, and neural network implementation. Unfortunately, many students submitted incomplete or partially functional solutions, or submitted no work at all. (*Note: Submitting three different files rather than one did not affect your grades.*)

How did we mark?

- If the code produced outputs matching the provided screenshots and followed naming conventions (e.g., the table being called Sales, insert would not allow duplication), full marks were awarded.
- In cases with few errors, markers made corrections to check functionality and awarded partial credit.
- For submissions with numerous errors, minimal marks were given for attempts.

The average mark on the Python question was 20.4/40.

Key issues by Python Task

While there were many different issues, here are a few of them.

Task 1: Python Basics and Data Manipulation [13 Marks]

Missing value handling: Many submissions did not appropriately handle missing values or apply suitable strategies for different column types (despite the example provided on Moodle).

Data conversion: Errors were frequently seen in converting dates correctly to a usable format, which affected subsequent tasks.

Plots: Bar and line plots often lacked correct formatting or appropriate aggregation of data (e.g., grouping by specific time periods like months).

Task 2: Python Database Management [15 marks]

Table creation: Some submissions did not explicitly define the table schema or included incorrect data types and constraints (e.g., setting Date as a primary key or not considering duplicate entries).

Data insertion: Data was either not inserted properly, lacked dynamic handling of the cleaned DataFrame, or failed to prevent duplicates.

Queries: Many query implementations were incomplete or failed due to schema inconsistencies or incorrect logic.

Task 3: Basic Neural Network Implementation [12 marks]

Model architecture: Several submissions did not follow the specified model architecture, including incorrect input/output sizes, missing activation functions, or inappropriate use of deprecated methods like `F.sigmoid`.

Data generation: Synthetic data generation was often absent or incorrect, with mismatched dimensions between inputs and targets.

Training and loss visualisation: While some submissions implemented training loops, issues with data loaders, loss calculations, or visualisation of training progress (e.g., plotting loss over epochs) were common, leading to no output.

Java: The Java exam required answering three different tasks, each in its own Java source file, and each file was marked manually for correctness as indicated below. For all tasks, no marks were allocated if students submitted only an empty class definition, or only a method header with no body.

We are aware that some students experienced an issue where their Java solution appeared not to save when they went to submit it. We believe that this was caused by students downloading the starter code zip file and accidentally editing the files in the zip file without extracting them, which would mean that their changes would not be saved to the classes involved in Task 1.

To address this issue, we made the following modification to the marking:

For any student who received 0 marks for Java Task 1, their mark for the Java question was based on their answers to Tasks 2 and 3 only (scaled up to a mark out of 40).

The average mark on the Java question was 17.9/40

Task 1: Representing an animal (10 marks)

Marks were deducted for issues including:

- Not using appropriate data types or access modifiers on the class fields: the name should be a string, the size should use the provided `AnimalSize`, and the comfortable temperature should be represented as a range of two integers.
- Not implementing all required getters/setters/equals/toString/hashCode methods
- Not correctly validating the fields
- Including validation in only the constructor or only the setters (should be in both)

Task 2: Enclosures and checks (15 marks)

Marks were deducted for issues including:

- Not including the correct fields in the Enclosure class – especially, the occupant should be a single Animal.
- Not having a correct constructor or getter methods; including setters despite the specification saying that they should not be included.
- Not following the specification for checkCompatibility.
- Not using checkCompatibility inside addAnimal; not correctly adding or removing animals.

Task 3: The pet care service (15 marks)

Marks were deducted for issues including:

- Improper definition of PetCare service
- printEnclosure not working properly because Enclosure.toString() is not implemented
- allocateAnimal not following specification, especially not checking the cost properly
- removeAnimal not following specification