

20. Further Programming

Starter Quiz · 课前小测

A-Level Computer Science

Name: _____ Score: _____

1. Which are true of low-level programming? Select **all** that apply.

- ☐ it gives direct access to registers and memory addresses
- ☐ it offers maximum control and speed and the smallest code
- ☐ it is architecture-specific, so it must be rewritten for another processor
- ☐ it is easy to write and maintain

Tick every correct option.

2. An object combines:

- ☐ attributes (data) and methods (operations)
- ☐ only data
- ☐ only functions
- ☐ two separate programs

3. What does opening a file FOR WRITE do to existing contents?

- ☐ it overwrites them (the file is created/emptied)
- ☐ it adds to the end
- ☐ it leaves them and reads only
- ☐ it encrypts them

4. An exception is:

- ☐ an error or unexpected condition that occurs while a program runs
- ☐ a syntax mistake caught at compile time
- ☐ a comment in the code
- ☐ a type of variable

5. Imperative (procedural) programming is based on:

- ☐ a sequence of commands that change the program's state
- ☐ stating facts and rules for an engine to infer
- ☐ objects that combine data and methods
- ☐ pure functions with no side effects

6. What is the relationship between a class and an object?

- ☐ the class is the blueprint; an object is an instance of it with its own values

- ☐ they are two words for the same thing
- ☐ an object is the blueprint for a class
- ☐ a class is a group of objects stored together

7. Why use exception handling rather than checking for every error in advance? Select **all** that apply.

- ☐ some errors cannot be prevented in advance, because conditions change between the check and the action
- ☐ otherwise every operation would need its own IF check, burying the algorithm
- ☐ it separates the normal flow from the error handling
- ☐ it makes the program run faster

Tick every correct option.

8. Why does a file search loop use WHILE NOT EOF AND NOT found? Select **all** that apply.

- ☐ so the loop stops as soon as the record is found instead of reading the rest
- ☐ so there is a flag to test afterwards to report "not found"
- ☐ so the loop does not read past the end of the file
- ☐ so the file does not need closing

Tick every correct option.

9. The block that runs whether or not an exception occurred, making it right for closing files, is _____.

10. Declarative paradigms (functional, logic, SQL) state WHAT to compute and let the runtime decide how, whereas imperative code spells out every step.

- ☐ True ☐ False

1. **it gives direct access to registers and memory addresses; it offers maximum control and speed and the smallest code; it is architecture-specific, so it must be rewritten for another processor**

Control is bought with effort: assembly is slow to write and hard to maintain, which is why it is confined to drivers and firmware.

2. **attributes (data) and methods (operations)**

An object bundles its data (attributes) with the operations (methods) that act on it.

3. **it overwrites them (the file is created/emptied)**

WRITE creates or overwrites the file. To keep existing contents and add more, use APPEND.

4. **an error or unexpected condition that occurs while a program runs**

Exceptions are run-time problems (divide by zero, file not found) that handling lets you respond to gracefully.

5. **a sequence of commands that change the program's state**

Imperative code gives step-by-step commands (assignments, loops, calls) that change state.

6. **the class is the blueprint; an object is an instance of it with its own values**

One Taxi class defines the attributes and methods; each actual taxi is an object with its own capacity and registration.

7. **some errors cannot be prevented in advance, because conditions change between the check and the action; otherwise every operation would need its own IF check, burying the algorithm; it separates the normal flow from the error handling**

The benefits are correctness and clarity. Speed is not one of them; handling adds a little overhead, which is worth paying.

8. **so the loop stops as soon as the record is found instead of reading the rest; so there is a flag to test afterwards to report "not found"; so the loop does not read past the end of the file**

Every file must still be closed. The two conditions handle early exit and the end of the file respectively.

9. **FINALLY**

Cleanup must happen on both paths. Putting CLOSEFILE only in the TRY block means it is skipped exactly when an error left the file open.

10. **True**

A SQL query says which rows you want, not how to scan the tables — the opposite of step-by-step imperative code.