

## 2.1 COMPUTATIONAL THINKING · 2.1.4 &amp; 2.1.5

# Thinking logically & concurrently

Decisions that control flow + tasks that can run at once. Spec 2.1.4–2.1.5.

## 01 Thinking logically (2.1.4)

**Point**      **Where** a choice is made (IF/CASE; flowchart diamond).

**Condition** The **test** deciding the path, e.g. age >= 18.

**Flow**      True/False → different statements run or are skipped.

## 02 Thinking concurrently (2.1.5)

**Def**      Run more than one task **at the same time**.

**Which**      Only **independent** parts (no shared dependency).

**Sequential** Steps needing a previous result stay in order.

## 03 Concurrency: benefits

**Faster**      Finishes sooner (esp. multi-core).

**Responsive** One slow task doesn't freeze the rest.

**Big jobs**      Weather, image processing, simulations.

## 04 Concurrency: trade-offs

**Complex**      Harder to design & debug.

**Clashes**      Shared data → race conditions.

**Limits**      Dependencies can't be parallelised.

**Overhead**      Speed-up **less than proportional**.

## FINAL PASS BEFORE THE EXAM

## Rapid exam tips

Six slips on logical/concurrent questions.

**01**

Decision **point** = where; **condition** = the test. Don't confuse them.

**02**

"How does it affect flow?" → describe **both** the True and False paths.

**03**

Only **independent** parts can run concurrently.

**04**

Concurrency essay = **benefits AND trade-offs** + a conclusion.

**05**

Speed-up is **less than proportional** — overhead & dependencies.

**06**

Good example: weather model — concurrent within a step, sequential between.