

Public PYQ Patterns: Research Synthesis

Pseudocode / programming logic

The public CoCubes sample PDF contains pseudocode/output questions that exercise:

- accumulating a loop total;
- repeated multiplication/power;
- loop bounds;
- number reversal using MOD and integer division;
- nested IF/ELSE binding;
- Fibonacci-like update order;
- arithmetic swapping;
- recursive sum and factorial.

High-return conclusion

Do not spend time memorising language syntax. Spend it on [[04_Pseudocode/Dry_Run_Method|variable tables]], MOD/DIV, recursion and update order.

Technical MCQ patterns

A publicly archived CoCubes-domain page reports questions around:

- DBMS transaction conflict serializability;
- page faults;
- hamming distance;
- IPv4 subnetting/host counts;
- channel-capacity calculation;
- valid host address selection.

Candidate reports from CoCubes-hosted tests also mention:

- DNS port, TCP vs UDP, HTTP methods/status;
- quicksort worst-case complexity;
- tree-level node count;
- queue/FIFO;
- binary search/divide-and-conquer;
- linked-list loop detection;
- SQL joins, count and transactions.

High-return conclusion

Technical questions are commonly textbook-level but time-limited. Learn the keyword-to-answer mappings, then practise a few application questions.

Coding patterns

A 2025 candidate report for a CoCubes coding assessment mentions:

- unit digit of a power;
- accumulating values from a collection.

Other public experiences show arrays/strings, tree properties and medium data-structure questions. That supports this priority:

1. number logic and modulo cycles;
2. linear scans;
3. strings/arrays;
4. hash maps and sorting;
5. basic trees/graphs only after the above.

Aptitude patterns

Public CoCubes-oriented material repeatedly lists:

- percentage, ratio/proportion, speed/time/distance;
- grammar correction, vocabulary and reading comprehension;
- series, coding-decoding, blood relations and statement/conclusion.

Gamified assessment

Official Aon guidance names:

- working memory/grid challenge;
- complex planning/motion challenge;
- deductive reasoning/switch challenge;
- numeracy/digit challenge.

This is why the vault gives a separate method for each ability instead of generic "play games" advice.

Revision use

Read [\[\[03_Technical/Technical_Quick_Recall|Technical quick recall\]\]](#), then solve [\[\[07_PYQ_Research/PYQ_Derived_Drills|PYQ-derived drills\]\]](#). Repeat any error with the linked concept note.

[\[\[07_PYQ_Research/PYQ_Research_Method|Research method\]\]](#) · [\[\[99_Sources|Sources\]\]](#)