

Research Conclusions: What to Prioritise

The short conclusion

Your supplied pattern has 80 timed MCQs before coding, plus a gamified assessment. The highest-return preparation is:

1. Pseudocode dry-running and programming logic.
2. Data structures, algorithms, OS, networking and quick factual technical topics.
3. Aptitude speed: percentages, ratios, averages, work/time/speed, grammar and standard reasoning.
4. Coding patterns for arrays, strings, number logic, sorting/searching and frequency counting.
5. Familiarity with the four Aon-style games; do not treat them as random games.

What public evidence shows

Evidence quality	Finding	Action
Official CoCubes sample material	Pseudocode samples use loop tracing, modulo, nested conditions, swaps and recursion.	Practise dry-running, not language syntax.
Public archived CoCubes domain paper	Reported topics include conflict serializability, page faults, hamming distance, subnetting and channel capacity.	Revise factual CS fundamentals and simple calculations.
Candidate reports from CoCubes-hosted tests	Reported questions include DNS, TCP vs UDP, HTTP, quicksort complexity, tree properties, queue, binary search and linked-list cycle detection.	Know definitions, use cases and Big-O.
Official Aon preparation page	The named games map to working memory, complex planning, deductive reasoning and numeracy.	Learn each game method and practise speed calmly.

Important limits

No public source can prove exact question weightage for your employer. The platform supports customised assessments, so company-specific instructions outrank all generic websites. Do not assume that an old report's time limits, language options, negative marking or elimination rules apply to you.

High-yield technical order

1. [\[\[03_Technical/Data_Structures|Data structures\]\]](#)
2. [\[\[03_Technical/Algorithms|Algorithms and complexity\]\]](#)
3. [\[\[03_Technical/Operating_Systems|Operating systems\]\]](#)

4. [\[\[03_Technical/Networking|Networking\]\]](#)
5. [\[\[03_Technical/Embedded_Systems|Basic embedded systems\]\]](#)
6. [\[\[03_Technical/Software_Development_Methodologies|SDLC and methodologies\]\]](#)
7. [\[\[03_Technical/Software_QA_and_Testing|QA and testing tools\]\]](#)

High-yield aptitude order

1. [\[\[02_Aptitude/Quant_Methods|Percentages, ratio, averages, time/work, speed/distance\]\]](#)
2. [\[\[02_Aptitude/English_Methods|Grammar, error spotting and reading comprehension\]\]](#)
3. [\[\[02_Aptitude/Reasoning_Methods|Series, direction, blood relations, syllogisms and arrangements\]\]](#)

[\[\[00_Start_Here/Dashboard|Back to dashboard\]\]](#) · [\[\[07_PYQ_Research/Public_PYQ_Patterns|Detailed PYQ patterns\]\]](#)