
tags: - cocubes - coding - methods

Coding Playbook

Two problems in thirty minutes is an execution test. The objective is not to show the most advanced algorithm you know; it is to submit accepted code with correct input/output and edge-case handling.

The five-minute start

For each problem:

1. Read input and output twice.
2. Write one custom example by hand.
3. Identify constraints. They decide whether $O(n^2)$, $O(n \log n)$ or $O(n)$ is needed.
4. List edge cases: empty/small input, one element, duplicates, negative values, zero, large values.
5. Choose the simplest correct pattern from `[[05_Coding/Pattern_Catalog]pattern catalog]]`.

Priority when two problems appear

1. Select the problem with a clear method immediately.
2. Submit a complete solution for it.
3. Test it on normal and boundary input.
4. Start the second problem.
5. If the second is hard, write the simpler correct approach if it can pass partial tests.

Never spend the whole section trying to derive one difficult solution before submitting anything.

Universal code checklist

- ☐ Correct function/class signature required by the platform.
- ☐ Read every input value, in correct order.
- ☐ Print exactly the expected output, no explanatory text.
- ☐ Use a sufficiently wide integer type for sums/products.
- ☐ Reset per-test-case variables.
- ☐ Handle duplicates and boundary indices.
- ☐ Check divide-by-zero and empty collection cases where relevant.
- ☐ Run sample tests, then invent at least two more.

Debug in this order

1. Input parsing.
2. Index boundaries: $0, n-1, \text{left} \leq \text{right}$.
3. Initialisation of answer/maximum/minimum.

4. Update order inside loop.
5. Output format.
6. Complexity.

When no language is specified

Use the language you can write, debug and explain most confidently. Do not switch to a new language in the final two days. Confirm the exact languages allowed in your test invitation before practising syntax.

[\[\[05_Coding/Pattern_Catalog|Pattern catalog\]\]](#) · [\[\[05_Coding/Coding_Problems|Practice problems\]\]](#) · [\[\[05_Coding/Language_Quick_Reference|Language reference\]\]](#)