# 🧠 Bloomberg Coding Interview Cheat Sheet

## ✅ Quick Strategy Checklist

* ☐ Restate the problem in your own words
* ☐ Ask clarifying questions (input range? duplicates? sorted?)
* ☐ Walk through small test case on paper
* ☐ Start with a brute force approach
* ☐ Think about possible patterns: hashmap? two pointers? sliding window?
* ☐ Write pseudocode if unsure
* ☐ Code clearly with clean variable names
* ☐ Test with normal, edge, and empty inputs
* ☐ State time and space complexity at the end

## 🧩 Common Patterns to Try

➡️ Hashing (unordered\_map / set)

➡️ Two Pointers

➡️ Sliding Window

➡️ Binary Search

➡️ Stack / Monotonic Stack

➡️ DFS / BFS for Graphs

➡️ Heap / Priority Queue

➡️ Dynamic Programming (build bottom-up)

➡️ Backtracking (combinations, permutations)

➡️ Prefix Sum or Difference Arrays

## ⏱ Time Complexity Guide

🕒 O(1): Constant time (hash map insert/lookup)

🕒 O(log n): Binary search

🕒 O(n): Single pass through array or map

🕒 O(n log n): Efficient sorting

🕒 O(n^2): Nested loops (brute force pair check)

🕒 O(2^n): Recursion/backtracking (worst-case)

🕒 O(n!): Permutations/combinations

## ⚠️ Common Edge Cases to Check

🔍 Empty input

🔍 Single element

🔍 All elements are the same

🔍 All elements are unique

🔍 Sorted vs unsorted input

🔍 Negative numbers

🔍 Very large inputs