Coding Mastery Phase Checklist

	Phase	Topic	Sub Topic
☐	Phase 1	Syntax & Fluency	Comfortable with loops and conditionals implementation
☐	Phase 1	Syntax & Fluency	Able to solve array traversal problems efficiently
☐	Phase 1	Syntax & Fluency	String manipulation problems solved without difficulty
☐	Phase 1	Syntax & Fluency	Mathematical implementation problems completed
☐	Phase 1	Syntax & Fluency	Basic brute-force problems implemented correctly
☐	Phase 1	Syntax & Fluency	Able to reduce simple brute-force solutions using observations
☐	Phase 1	Syntax & Fluency	Understand time complexity difference between $O(n^2)$ and $O(n)$
☐	Phase 1	Syntax & Fluency	Bit manipulation basics practiced
☐	Phase 1	Syntax & Fluency	Mathematical optimization problems attempted
☐	Phase 1	Syntax & Fluency	Coding speed improved (Easy problems solved <15 min)
☐	Phase 2	Must-Know Logic	Prefix sum techniques applied in problems
☐	Phase 2	Must-Know Logic	Hashmap frequency counting problems completed
☐	Phase 2	Must-Know Logic	Pair and triplet counting optimized using hashing
☐	Phase 2	Must-Know Logic	Rolling frequency / sliding counting problems solved
☐	Phase 2	Must-Know Logic	Pattern observation problems practiced
☐	Phase 2	Must-Know Logic	Prefix-hash transformations implemented
☐	Phase 2	Must-Know Logic	Counting problems optimized from brute-force to O(n)
☐	Phase 2	Must-Know Logic	Multi-map hashing problems attempted
☐	Phase 2	Must-Know Logic	Combinatorial counting logic problems solved
☐	Phase 2	Must-Know Logic	Advanced counting problems using hashmap completed
☐	Phase 3	Algorithmic Thinking	Two-pointer pattern problems mastered
☐	Phase 3	Algorithmic Thinking	Sliding window problems (basic → advanced) solved
☐	Phase 3	Algorithmic Thinking	Binary search applications practiced
☐	Phase 3	Algorithmic Thinking	Binary search on answer problems solved
☐	Phase 3	Algorithmic Thinking	Greedy algorithm reasoning understood
☐	Phase 3	Algorithmic Thinking	Recursion problems implemented correctly
☐	Phase 3	Algorithmic Thinking	Recursion pruning techniques practiced
☐	Phase 3	Algorithmic Thinking	Sorting-based optimization problems completed
☐	Phase 3	Algorithmic Thinking	Multi-constraint optimization problems solved
☐	Phase 3	Algorithmic Thinking	Medium–hard algorithmic problems attempted confidently
☐	Phase 4	Linear DSA	Linked list traversal and manipulation problems solved
☐	Phase 4	Linear DSA	Multi-pointer linked list problems practiced
☐	Phase 4	Linear DSA	Stack-based problems mastered
☐	Phase 4	Linear DSA	Queue and circular queue problems solved
☐	Phase 4	Linear DSA	Monotonic stack pattern problems practiced
☐	Phase 4	Linear DSA	Heap / priority queue problems solved
☐	Phase 4	Linear DSA	Heap + greedy hybrid problems attempted
☐	Phase 4	Linear DSA	Stack/queue design problems implemented
☐	Phase 4	Linear DSA	Advanced DS design problems attempted
☐	Phase 4	Linear DSA	Linear data structure patterns recognized quickly
☐	Phase 5	Non-Linear DSA	Binary tree traversal problems mastered
☐	Phase 5	Non-Linear DSA	Tree recursion problems solved confidently
☐	Phase 5	Non-Linear DSA	Tree DP introduction problems practiced
☐	Phase 5	Non-Linear DSA	Binary search tree problems completed
☐	Phase 5	Non-Linear DSA	Trie implementation and usage practiced
☐	Phase 5	Non-Linear DSA	Graph BFS and DFS problems solved
☐	Phase 5	Non-Linear DSA	Graph connectivity problems (Union-Find) attempted
☐	Phase 5	Non-Linear DSA	State-based BFS/DFS problems solved
☐	Phase 5	Non-Linear DSA	Tree rerooting / transformation problems practiced
☐	Phase 5	Non-Linear DSA	Structural recursion reasoning improved
☐	Phase 6	Advanced Algorithms	Greedy advanced scheduling problems solved
☐	Phase 6	Advanced Algorithms	Dynamic programming patterns identified and applied
☐	Phase 6	Advanced Algorithms	Hard DP problems attempted
☐	Phase 6	Advanced Algorithms	Backtracking with pruning problems solved
☐	Phase 6	Advanced Algorithms	Divide-and-conquer problems implemented
☐	Phase 6	Advanced Algorithms	Advanced shortest-path graph problems practiced
☐	Phase 6	Advanced Algorithms	Advanced string algorithm problems attempted
☐	Phase 6	Advanced Algorithms	Rolling hash / pattern matching problems solved
☐	Phase 6	Advanced Algorithms	Multi-concept integrated problems attempted
☐	Phase 6	Advanced Algorithms	Hard interview-level problems solved independently