QUEUE + DEQUE + PRIORITY QUEUE ROADMAP

EASY LEVEL (1–15)

---

1. Implement Queue using Array — Circular array
2. Implement Queue using Linked List — LL
3. Queue using Two Stacks — Stack simulation
4. Stack using Two Queues — Queue simulation
5. Circular Queue Implementation — Array (mod math)
6. Deque Implementation — Array/LL
7. Priority Queue Implementation — Heap
8. Queue using Java Queue
9. Deque using Java Deque
10. Generate binary numbers using queue
11. Reverse a queue — Recursion/stack
12. Interleave first and second half of queue
13. First non-repeating character in stream
14. Level order traversal — BFS
15. Check leaves at same level — BFS

MEDIUM LEVEL (16–35)

---

16. Sliding Window Maximum — Monotonic deque
17. Sliding Window Minimum — Monotonic deque
18. First negative integer in every window — Queue
19. Count distinct elements in windows — Hash + queue
20. Sum of min/max of every window — Deque
21. Max of all subarrays size K — Deque
22. Design Hit Counter — Queue
23. Circular Deque — Deque ops
24. Front-Middle-Back Queue
25. Moving average from stream — Queue
26. LRU Cache — DLL + hashmap

27. LFU Cache — Multi-queue + map
28. Snake and Ladder — BFS
29. Rotting Oranges — BFS
30. Walls and Gates — BFS
31. Number of islands (online) — BFS
32. Course Schedule I — Kahn’s Algo
33. Course Schedule II — Queue topo
34. Detect cycle in directed graph — BFS
35. Minimum depth of binary tree — BFS

HARD LEVEL (36–55)

---

36. Shortest path in binary matrix — BFS
37. Word Ladder — BFS
38. Word Ladder II — BFS + backtracking
39. 0-1 BFS — Deque
40. Dijkstra using PQ
41. Prim’s MST using PQ
42. Merge K sorted lists — Min heap
43. K closest points — Heap
44. Kth largest in stream — Heap
45. Median in data stream — Two heaps
46. Sliding window median — Two heaps
47. Top K frequent elements — Heap
48. Rearrange string no same adjacent — Max heap
49. Task Scheduler — Max heap + queue
50. Maximum score from cards — Deque
51. Monotonic queue DP optimization
52. Multi-level cache design
53. Real-time job scheduler
54. Time-based key-value store
55. Producer–Consumer (queue)

BONUS (56–60)

---

56. PQ with custom comparator (Java)
57. PriorityBlockingQueue usage
58. Queue vs Deque vs ConcurrentLinkedQueue
59. Bounded queue with wait/notify
60. Rate limiter using sliding window queue