HCL previous year — 200 Practice Questions (HackerRank) By – Durgesh StudyHub

50 Coding Questions | Problem + Approach (HCL / Product-based style)

✨ HARD LEVEL QUESTIONS (1–50)

1. Trapping Rain Water

Problem: Find total water trapped between bars after rain.
Approach: Two-pointer or prefix-max & suffix-max arrays.

2. Largest Rectangle in Histogram

Problem: Largest rectangle area possible from heights.
Approach: Monotonic stack maintaining increasing heights.

3. Minimum Window Substring

Problem: Smallest substring that contains all characters of target string.
Approach: Sliding-window + frequency hashmap; shrink while valid.

4. Longest Substring with K Distinct Characters

Problem: Longest window having ≤ K unique characters.
Approach: Sliding-window + hashmap remove when size > K.

---

5. Count Subarrays With Sum = K

Problem: Count number of subarrays whose sum equals K.
Approach: Prefix-sum + hashmap count.

---

6. Median of Two Sorted Arrays

Problem: Find median in O(log n) without merging arrays.
Approach: Binary search partition on small array.

---

7. Word Ladder

Problem: Convert start word to end word using dictionary steps.
Approach: BFS shortest transformation path.

---

8. LRU Cache

Problem: Implement cache with O(1) get & put.
Approach: Doubly linked list + Hashmap.

---

9. Longest Increasing Path in Matrix

Problem: Maximum length increasing path in grid.
Approach: DFS + DP memoization.

---

10. Minimum Number of Platforms

Problem: Minimum platforms needed for trains to not overlap.
Approach: Sort arrivals & departures, two-pointer sweep.

---

11. Maximum Product Subarray

Problem: Max product of contiguous elements.
Approach: Track current min & max due to negative flips.

---

12. Count Inversions in Array

Problem: Number of pairs i < j and A[i] > A[j].
Approach: Merge sort with inversion count.

---

13. N-Queens

Problem: Place N queens so none attack each other.
Approach: Backtracking + column/diag checks.

---

14. Rat in a Maze

Problem: Print all possible valid paths to destination.
Approach: Backtracking movement.

---

15. Sudoku Solver

Problem: Solve 9×9 Sudoku grid.
Approach: Backtracking + validity constraints.

---

16. Generate All Permutations

Problem: Print all permutations of array or string.
Approach: Backtracking swap positions.

---

17. Edit Distance

Problem: Minimum edits to convert A to B.
Approach: Matrix DP using replace/insert/delete.

---

18. Burst Balloons

Problem: Max coins gained from balloon bursting.
Approach: Interval DP.

---

19. Maximum Sum Increasing Subsequence

Problem: Max sum of increasing subsequence.
Approach: DP $O(n^2)$ or segment tree.

---

20. Longest Common Subsequence

Problem: Largest subsequence present in both strings.
Approach: DP table m×n.

---

21. Job Sequencing with Deadlines

Problem: Max profit by selecting jobs under deadlines.
Approach: Sort by profit + DSU or greedy slotting.

---

22. Russian Doll Envelopes

Problem: Max envelopes that can fit inside each other.
Approach: Sort + LIS on width.

---

23. K Closest Points to Origin

Problem: Return k smallest Euclidean distances.
Approach: Max-heap of size k.

---

24. Find Bridges in Graph

Problem: Bridges whose removal disconnects graph.
Approach: Tarjan DFS discovery & low time.

---

25. Topological Sorting + Cycle Detection

Problem: Order tasks to be processed without violating dependencies.
Approach: Kahn’s BFS or DFS with visited recursion.

---

26. Maximum XOR Pair

Problem: Max XOR of pair from array.
Approach: Trie bitwise storing binary of numbers.

---

27. Reverse Nodes in K-Group (Linked List)

Problem: Reverse every K consecutive nodes.
Approach: Pointer iteration + repeated reversing.

---

28. Vertical Order Traversal (Tree)

Problem: Group tree nodes column-wise.
Approach: BFS with coordinates mapping.

---

29. Serialize and Deserialize Binary Tree

Problem: Convert tree to string and back.
Approach: Preorder + NULL markers.

---

30. Binary Tree to Doubly Linked List

Problem: Convert tree to in-order DLL.
Approach: Inorder traversal linking nodes.

---

31. Kth Smallest Element in BST

Problem: Find Kth smallest element.
Approach: Inorder traversal counting.

---

32. Maximum Sum Rectangle in Matrix

Problem: Max sum rectangle anywhere in grid.
Approach: Kadane on compressed columns.

---

33. Aggressive Cows

Problem: Maximize minimum distance between cows.
Approach: Binary search on distance.

---

34. Painter’s Partition

Problem: Min time to paint given boards by k painters.
Approach: Binary search partition.

---

35. Book Allocation Problem

Problem: Min max pages for k students.
Approach: Binary search on answer.

---

36. Celebrity Problem

Problem: Find person known by everyone but knows nobody.
Approach: Two-pointer elimination.

---

37. Flatten Nested List Iterator

Problem: Iterate list containing nested lists.
Approach: Stack flatten levels.

---

38. Longest Chain of Pairs

Problem: Max linkable ordered pairs.
Approach: Greedy based on second value.

---

39. Course Schedule

Problem: Check if course schedule valid with prerequisites.
Approach: Topological sort.

---

40. Knight Shortest Path

Problem: Minimum steps chess knight takes.
Approach: BFS.

---

41. Number of Islands

Problem: Count connected components in grid.
Approach: DFS/BFS marking visited.

---

42. Autocomplete System

Problem: Suggest top strings using prefix search.
Approach: Trie + priority ordering.

---

43. Grid Minimum Path Sum

Problem: Minimum cost route.
Approach: DP.

---

44. Minimum Spanning Tree

Problem: MST in weighted graph.
Approach: Kruskal + DSU or Prim.

---

45. Bellman Ford

Problem: Shortest path with negative edges.
Approach: Relax edges N−1 times.

---

46. Articulation Points

Problem: Critical nodes whose removal splits network.
Approach: Tarjan DFS time-low array.

---

47. Fenwick Tree / BIT

Problem: Prefix-sum queries and updates.
Approach: Binary indexed structure.

---

48. Segment Tree + Lazy Propagation

Problem: Range update and query.
Approach: Segment tree with deferred updates.

---

49. Kth Smallest in Sorted Matrix

Problem: Sorted row/column matrix query.
Approach: Min-heap or binary search.

---

50. Sliding Window Maximum

Problem: Max of every window size K.
Approach: Deque storing useful elements.

---

1. Move all ‘#’ to the front

Problem: Rearrange string to move all # to start, keeping order of others.
Approach: Count #, rebuild string OR two-pointer stable move.

---

2. Move all zeros to the end

Problem: Move all 0s to end maintaining relative order of non-zero elements.
Approach: Two-pointer write index, O(n) in-place.

---

3. Smallest subarray with sum ≥ K

Problem: Find minimum size subarray with sum ≥ K.
Approach: Sliding-window expand & shrink, O(n).

---

4. Longest substring without repeating characters

Problem: Return longest substring with unique characters.
Approach: Sliding-window + map storing last index.

---

5. Count distinct in every window of size K

Problem: For each sliding window, print count of distinct elements.
Approach: Hashmap frequency sliding window.

---

6. Merge overlapping intervals

Problem: Merge intervals to remove overlaps.
Approach: Sort by start, then merge sequentially.

---

7. Kth largest element

Problem: Find Kth largest in array.
Approach: Min-heap of size K OR quickselect.

8. Check if strings are anagrams

Problem: Check if 2 strings have same characters & freq.
Approach: Frequency array 26 or hashmap.

9. First non-repeating character

Problem: First character whose freq = 1.
Approach: Frequency count + second scan.

10. Balanced brackets

Problem: Validate parentheses/brackets/braces.
Approach: Stack push-pop matching.

11. Rotate array by K

Problem: Rotate right or left by K positions.
Approach: Reverse three parts OR extra array method.

12. Missing number from 1 to N

Problem: One number missing from 1…N.
Approach: XOR method or sum formula.

13. Majority element (appearing > n/2)

Problem: Find element appearing more than half.
Approach: Boyer–Moore Voting Algorithm.

14. Product of array except itself

Problem: Result[i] = product of all others.
Approach: Prefix * suffix product arrays.

15. Leaders in array

Problem: Elements greater than elements to right.
Approach: Traverse from right, track max.

16. Stock buy-sell profit

Problem: Max profit from buying & selling once.
Approach: Track min so far + best difference.

17. Next greater element

Problem: For each element find nearest greater to right.
Approach: Stack decreasing order.

18. Find duplicate number

Problem: A number repeats in array 1…n.
Approach: Floyd cycle detection (linked list style).

19. Sort characters by frequency

Problem: Characters sorted by decreasing freq.
Approach: Map + heap.

---

20. Check palindrome string

Problem: Return true if string same forwards/backwards.
Approach: Two-pointers from ends.

---

21. Remove duplicates from sorted array

Problem: Remove duplicates in-place return new length.
Approach: Two-pointer write next unique.

---

22. Binary search

Problem: Find element in sorted array.
Approach: Divide mid & search.

---

23. Find peak element

Problem: Element greater than neighbors.
Approach: Binary search mountain.

---

24. Count pairs sum = K

Problem: Count pairs whose sum equals K.
Approach: Hashmap count.

---

25. Reorder string by alphabet position

Problem: Sort string alphabetically.
Approach: Count sort.

26. Remove all adjacent duplicates

Problem: Remove duplicates that appear consecutively.
Approach: Stack compare last inserted.

27. Reverse words in a sentence

Problem: Reverse word order.
Approach: Split + reverse.

28. Longest palindrome substring

Problem: Longest substring which is palindrome.
Approach: Expand around center.

29. Max consecutive ones

Problem: Longest run of 1s.
Approach: Count continuous segments.

30. Count vowels & consonants

Problem: Return count of vowels & consonants.
Approach: Traversal + check sets.

31. Check subsequence

Problem: Check if string A is subsequence of B.
Approach: Two pointers.

32. Rearrange +ve & -ve numbers alternating

Problem: Alternate positive & negative values.
Approach: Partition + merge.

33. Spiral print matrix

Problem: Print matrix elements spiral order.
Approach: Boundary simulation.

34. Transpose matrix

Problem: Swap rows & columns.
Approach: Swap i,j indexes.

35. Max sum subarray (Kadane)

Problem: Largest sum contiguous subarray.
Approach: Kadane O(n).

36. Two sum

Problem: Return 2 indexes that sum to target.
Approach: Hashmap.

37. Generate Pascal triangle

Problem: Print N rows.
Approach: DP formula next = prev*(row-col)/col.

---

38. Check power of 2

Problem: Check if number is 2^n.
Approach: bitwise n & (n-1) == 0.

39. Count bits (number of 1s)

Problem: Count 1 bits in integer.
Approach: Brian Kernighan method.

40. Add binary strings

Problem: Add 2 binary representation strings.
Approach: Add from end carry logic.

---

41. GCD of two numbers

Problem: Return greatest common divisor.
Approach: Euclidean algorithm.

42. LCM of two numbers

Problem: Return least common multiple.
Approach: lcm(a,b) = a*b / gcd.

43. Reverse linked list

Problem: Reverse nodes.
Approach: Pointer prev-next.

44. Detect cycle linked list

Problem: Determine if cycle exists.
Approach: Floyd fast-slow pointers.

45. Level order traversal of tree

Problem: Print nodes level by level.
Approach: BFS queue.

46. Height of binary tree

Problem: Max depth of tree.
Approach: Recursive DFS.

47. Check BST

Problem: Validate binary search tree constraints.
Approach: Inorder check sorted.

48. Path sum root-to-leaf

Problem: Check if path sum equals target.
Approach: DFS accumulate path sum.

49. Shortest path BFS in graph

Problem: Min distance src → dest.
Approach: BFS queue visited.

50. Dijkstra shortest path

Problem: Weighted graph shortest path.
Approach: Min-heap + relax edges.

NEXT 100 HARD LEVEL QUESTIONS (Problem + Approach)

Hard Level (1–25)

1. Trapping Rain Water

Problem: Find trapped rainwater between building heights.
Approach: Prefix-max & suffix-max OR Two-pointer O(n).

2. Largest Rectangle in Histogram

Problem: Max area rectangle possible from bars.
Approach: Monotonic stack.

3. Minimum Window Substring

Problem: Smallest window containing all chars of target string.
Approach: Sliding-window + freq map.

4. Longest Substring with K distinct characters

Problem: Longest window with ≤ K unique.
Approach: Sliding-window + hashmap.

5. Count subarrays with sum = K

Problem: Total subarrays whose sum equals K.
Approach: Prefix sum + hashmap.

6. Median of two sorted arrays

Problem: Find median without merging.
Approach: Binary search partition.

7. Word Ladder

Problem: Transform word to ladder using dictionary steps.
Approach: BFS + adjacency pattern.

8. LRU Cache

Problem: Implement Least Recently Used cache.
Approach: DLL + Hashmap.

9. Longest Increasing Path in Matrix

Problem: Max path increasing value moves.
Approach: DFS + DP + memo.

10. Minimum number of Platforms

Problem: Train arrival departure schedule.
Approach: Sort arrival + departure, sweep.

11. Maximum Product Subarray

Problem: Largest product contiguous sequence.
Approach: Track min & max.

12. Count Inversions in array

Problem: Pairs (i, j) where A[i] > A[j].
Approach: Merge sort counting.

---

13. N-Queens problem

Problem: Place queens without attack.
Approach: Backtracking bitsets.

---

14. Rat in a Maze

Problem: Print all possible paths.
Approach: Backtracking recursion.

---

15. Sudoku Solver

Problem: Solve 9×9 Sudoku board.
Approach: Backtracking + constraint checks.

---

16. Generate all permutations

Problem: Print all permutations of array/string.
Approach: Backtracking/swapping.

---

17. Edit Distance

Problem: Min insertion/deletion/substitution.
Approach: DP string DP.

---

18. Burst Balloons

Problem: Max coins from balloon bursting order.
Approach: Interval DP.

---

19. Maximum sum increasing subsequence

Problem: Max sum LIS excluding values.
Approach: DP O(n²) or segment tree.

---

20. Longest Common Subsequence

Problem: Max subsequence in both strings.
Approach: DP m×n.

---

21. Job sequencing with deadlines

Problem: Max profit scheduling.
Approach: Sort profit + DSU.

---

22. Russian Doll Envelopes

Problem: Max nest envelopes.
Approach: Sorting + LIS.

---

23. K closest points to origin

Problem: Min distance K elements.
Approach: Max heap size K.

---

24. Find bridges in graph

Problem: Bridges whose removal disconnect graph.
Approach: Tarjan DFS discovery time.

---

25. Topological sort + detect cycle

Problem: DAG ordering.
Approach: Kahn's BFS or DFS recursion stack.
