@CODERS_SECTION

100 DSA QUESTIONS

Arrays & Math (Advanced)

1. Maximum subarray sum in circular array
2. Median of two sorted arrays
3. Count inversions in an array
4. Maximum product subarray
5. Trapping rain water
6. Subarray with XOR = K
7. Find the repeating and missing number
8. Maximum XOR of two numbers in an array
9. Count number of subarrays with given sum
10. Longest subarray with 0 sum

Strings (Advanced)

1. Longest substring without repeating characters
2. Count distinct substrings using Trie
3. Longest common subsequence
4. Longest palindromic subsequence
5. Minimum insertions to make a string palindrome
6. Rabin-Karp algorithm
7. KMP (Prefix function)
8. Z-algorithm for pattern matching
9. String isomorphism
10. Smallest window containing all characters of another string

Linked List (Advanced)

1. Reverse nodes in k-group
2. Flatten a multilevel linked list
3. Copy list with random pointer (Deep Copy)
4. Merge k sorted linked lists
5. Detect and remove loop
6. Add two numbers represented by linked lists
7. Sort a linked list in O(n log n) time
8. Find intersection point of two Y-shaped linked lists
9. Find length of loop in linked list
10. Clone a linked list with next and random pointer

Stack & Queue (Advanced)

- Largest rectangle in histogram
- Sliding window maximum
- Implement LFU cache
- Design a stack with getMin in O(1)
- Expression evaluation (infix to postfix, evaluation)
- Maximum area in binary matrix
- Decode string with nested encoding
- Remove k digits to get smallest number
- Online stock span
- Build a min-stack with constant space

Trees (Advanced)

1. Serialize and deserialize binary tree
2. Construct tree from inorder and preorder
3. Diameter of binary tree
4. Kth ancestor of a node
5. Morris Inorder Traversal (O(1) space)
6. Lowest Common Ancestor (LCA)
7. Binary Tree to DLL
8. Vertical order traversal
9. Distance between two nodes
10. Boundary traversal of binary tree

Binary Search Trees (Advanced)

1. Convert BST to a Greater Tree
2. Recover BST where two nodes are swapped
3. Find kth smallest/largest in BST
4. BST iterator implementation
5. Merge two BSTs
6. Validate BST with duplicates
7. Inorder predecessor and successor in BST
8. Count nodes in complete binary tree ($\log^2 N$)
9. Range sum of BST
10. Largest BST in a Binary Tree

Recursion & Backtracking (Advanced)

- N-Queens
- Sudoku solver
- Word search II (with Trie)
- Combination sum with constraints
- All unique permutations with duplicates
- Expression add operators
- Maximum score words formed by letters
- Restore IP addresses
- Partition to k equal sum subsets
- Palindrome partitioning with minimum cuts

Dynamic Programming (Advanced)

- Edit distance
- Wildcard pattern matching
- 0/1 Knapsack
- Matrix chain multiplication
- Longest increasing subsequence (O(n log n))
- Count all palindromic substrings
- Max profit in job scheduling
- Minimum jumps to reach end
- Longest bitonic subsequence
- Egg dropping puzzle

Comment "**DSA**" to get this In PDF

@CODERS_SECTION

100 DSA QUESTIONS

Graphs (Advanced)

- Dijkstra's algorithm
- Bellman-Ford algorithm
- Floyd Warshall (All-pairs shortest path)
- Detect cycle in directed graph
- Topological sort (Kahn's algorithm)
- Kosaraju's algorithm (SCC)
- Kruskal's algorithm (Minimum Spanning Tree)
- Prim's algorithm
- Articulation points and bridges
- Word ladder (Shortest transformation sequence)

Tries & Hashing (Advanced)

1. Implement Trie with insert/search/delete
2. Maximum XOR pair in an array
3. Count words with given prefix
4. Auto-suggestion system using Trie
5. Implement Magic Dictionary
6. Longest word with all prefixes
7. Design Add and Search Word Data Structure
8. Substring with at most K distinct characters
9. Group anagrams using hashing

Count pairs with given XOR

Check Out Story Highlights for more Cheatsheets and coding Stuffs.

Like Comment Share Save

Comment "**DSA**" to get this In PDF