

DSA by Shradha Didi & Aman Bhaiya			
	<a href="#">Meet us on Youtube (Apna College)</a>		
Easy	Ideal Time : 5-10 mins		
Medium	Ideal Time : 15-20 mins	5 Questions each Day	
Hard	Ideal Time : 40-60 mins (based on Qs)		
Topics	Question (375)	Companies	Remarks
Arrays	<a href="#">Maximum and Minimum Element in an Array</a>		
Arrays	<a href="#">Reverse the Array</a>		
Arrays	<a href="#">Maximum-Subarray</a>	Microsoft + Facebook Interview Qs	<a href="#">use Kadane's Algorithm</a>
Arrays	<a href="#">Contains Duplicate</a>		
Arrays	<a href="#">Chocolate Distribution Problem</a>	Amazon Interview Qs	
Arrays	<a href="#">Search an Element in a Sorted and Pivoted Array</a>		
Arrays	<a href="#">Next Permutation</a>	Uber + Goldman Sachs + Adobe Interview Qs	
Arrays	<a href="#">Best time to Buy and Sell Stock</a>		
Arrays	<a href="#">Repeat and Missing Number Array</a>	Amazon Interview Qs	
Arrays	<a href="#">Kth-Largest Element in an Array</a>		
Arrays	<a href="#">Trapping Rain Water</a>	Samsung Interview Qs	
Arrays	<a href="#">Product of Array Except Self</a>	Microsoft + Facebook Interview Qs	
Arrays	<a href="#">Maximum Product Subarray</a>		
Arrays	<a href="#">Find Minimum in Rotated Sorted Array</a>		
Arrays	<a href="#">Search in Rotated Sorted Array</a>	Microsoft + Google + Apple Interview Qs	
Arrays	<a href="#">3Sum</a>		
Arrays	<a href="#">Container With Most Water</a>	Flipkart + Dunzo Interview Qs	
Arrays	<a href="#">Given Sum Pair</a>	Infosys + Amazon + Flipkart Interview Qs	
Arrays	<a href="#">Kth - Smallest Element</a>		
Arrays	<a href="#">Merge Overlapping Intervals</a>	Google Interview Qs	
Arrays	<a href="#">Find Minimum Number of Merge Operations to Make an Array Palindrome</a>		
Arrays	<a href="#">Given an Array of Numbers Arrange the Numbers to Form the Biggest Number</a>	Barclays Interview Qs	
Arrays	<a href="#">Space Optimization Using Bit Manipulations</a>		
Arrays	<a href="#">Subarray Sum Divisible K</a>		
Arrays	<a href="#">Print all Possible Combinations of r Elements in a Given Array of Size n</a>		
Arrays	<a href="#">Mo's Algorithm</a>		
Strings	<a href="#">Valid Palindrome</a>		
Strings	<a href="#">Valid Anagram</a>		
Strings	<a href="#">Valid parentheses</a>	Google Interview Qs	<a href="#">use Stacks (if possible)</a>
Strings	<a href="#">Remove Consecutive Characters</a>		
Strings	<a href="#">Longest Common Prefix</a>	Adobe + Grofers + Dunzo Interview Qs	
Strings	<a href="#">Convert a Sentence into its Equivalent Mobile Numeric Keypad Sequence</a>		
Strings	<a href="#">Print all the Duplicates in the Input String</a>	Ola + Amdocs IQ	
Strings	<a href="#">Longest Substring without Repeating Characters</a>	Morgan Stanley + Amazon IQ	
Strings	<a href="#">Longest Repeating Character Replacement</a>		
Strings	<a href="#">Group Anagrams</a>	Samsung + Adobe + Amazon Interview Qs	
Strings	<a href="#">Longest Palindromic Substring</a>	Microsoft + Google + Samsung + Visa IQ	
Strings	<a href="#">Palindromic Substrings</a>	Microsoft IQ	
Strings	<a href="#">Next Permutation</a>		
Strings	<a href="#">Count Palindromic Subsequences</a>	Myntra Interview Qs	
Strings	<a href="#">Smallest Window in a String Containing all the Characters of Another String</a>	Microsoft + Amazon IQ	
Strings	<a href="#">Wildcard String Matching</a>	Microsoft + Amazon + Ola IQ	
Strings	<a href="#">Longest Prefix Suffix</a>	Flipkart + Swiggy IQ	
Strings	<a href="#">Rabin-Karp Algorithm for Pattern Searching</a>		
Strings	<a href="#">Transform One String to Another using Minimum Number of Given Operation</a>		
Strings	<a href="#">Minimum Window Substring</a>		
Strings	<a href="#">Boyer Moore Algorithm for Pattern Searching</a>		
Strings	<a href="#">Word Wrap</a>		<a href="#">use Dynaming Programming</a>
2D Arrays	<a href="#">Zigzag (or diagonal) Traversal of Matrix</a>		
2D Arrays	<a href="#">Set Matrix Zeroes</a>		
2D Arrays	<a href="#">Spiral Matrix</a>	Flipkart + Apple + Societe Generale IQ	
2D Arrays	<a href="#">Rotate Image</a>		
2D Arrays	<a href="#">Word Search</a>	Google + Ola + Goldman Sachs IQ	
2D Arrays	<a href="#">Find the Number of Islands   Set 1 (Using DFS)</a>	Microsoft + Uber + Apple + Amazon IQ	<a href="#">Read about DFS</a>
2D Arrays	<a href="#">Given a Matrix of 'O' and 'X', Replace 'O' with 'X' if Surrounded by 'X'</a>		
2D Arrays	<a href="#">Find a Common Element in all Rows of a Given Row-Wise Sorted Matrix</a>		
2D Arrays	<a href="#">Create a Matrix with Alternating Rectangles of O and X</a>		
2D Arrays	<a href="#">Maximum Size Rectangle of all 1s</a>		
Searching & Sorting	<a href="#">Permute Two Arrays such that Sum of Every Pair is Greater or Equal to K</a>		
Searching & Sorting	<a href="#">counting sort</a>		
Searching & Sorting	<a href="#">find common elements three sorted arrays</a>		
Searching & Sorting	<a href="#">Searching in an array where adjacent differ by at most k</a>		
Searching & Sorting	<a href="#">ceiling in a sorted array</a>		
Searching & Sorting	<a href="#">Piar with given difference</a>		
Searching & Sorting	<a href="#">majority element</a>		
Searching & Sorting	<a href="#">count triplets with sum smaller that a given value</a>		
Searching & Sorting	<a href="#">Maximum Sum Subsequence with no adjacent elements</a>		
Searching & Sorting	<a href="#">Merge Sorted Arrays using O(1) Space</a>		
Searching & Sorting	<a href="#">Inversion of Array</a>		
Searching & Sorting	<a href="#">Find Duplicates in O(n) Time and O(1) Extra Space</a>		
Searching & Sorting	<a href="#">Radix Sort</a>		
Searching & Sorting	<a href="#">Product of Array except itself</a>		
Searching & Sorting	<a href="#">Make all Array Elements Equal</a>		

Searching & Sorting	<a href="#">Check if Reversing a Sub Array Make the Array Sorted</a>		
Searching & Sorting	<a href="#">Find Four Elements that Sum to a Given Value</a>		
Searching & Sorting	<a href="#">Median of Two Sorted Array with Different Size</a>		
Searching & Sorting	<a href="#">Median of Stream of Integers Running Integers</a>		
Searching & Sorting	<a href="#">Print Subarrays with 0 Sum</a>		
Searching & Sorting	<a href="#">Aggressive Cows</a>		
Searching & Sorting	<a href="#">Allocate Minimum number of Pages</a>		
Searching & Sorting	<a href="#">Minimum Swaps to Sort</a>		
Backtracking	<a href="#">Backtracking Set 2 Rat in a Maze</a>		
Backtracking	<a href="#">Combinational Sum</a>		
Backtracking	<a href="#">Crossword-Puzzle</a>		
Backtracking	<a href="#">Longest Possible Route in a Matrix with Hurdles</a>		
Backtracking	<a href="#">Printing all solutions in N-Queen Problem</a>		
Backtracking	<a href="#">Solve the Sudoku</a>		
Backtracking	<a href="#">Partition Equal Subset Sum</a>		
Backtracking	<a href="#">M Coloring Problem</a>		
Backtracking	<a href="#">Knight Tour</a>		
Backtracking	<a href="#">Soduko</a>		
Backtracking	<a href="#">Remove Invalid Parentheses</a>		
Backtracking	<a href="#">Word Break Problem using Backtracking</a>		
Backtracking	<a href="#">Print all Palindromic Partitions of a String</a>		
Backtracking	<a href="#">Find Shortest Safe Route in a Path with Landmines</a>		
Backtracking	<a href="#">Partition of Set into K Subsets with Equal Sum</a>		
Backtracking	<a href="#">Backtracking set-7 hamiltonian cycle</a>		
Backtracking	<a href="#">tug-of-war</a>		
Backtracking	<a href="#">Maximum Possible Number by doing at most K swaps</a>		
Backtracking	<a href="#">Backtracking set-8 solving cryptarithmic puzzles</a>		
Backtracking	<a href="#">Find paths from corner cell to middle cell in maze</a>		
Backtracking	<a href="#">Arithmetic Expressions</a>		
Linked List	<a href="#">Reverse Linked List</a>		
Linked List	<a href="#">Linked List Cycle</a>		
Linked List	<a href="#">Merge Two Sorted Lists</a>		
Linked List	<a href="#">Delete without Head node</a>		
Linked List	<a href="#">Remove duplicates from an unsorted linked list</a>		
Linked List	<a href="#">Sort a linked list of 0s-1s-or-2s</a>		
Linked List	<a href="#">Multiply two numbers represented linked lists</a>		
Linked List	<a href="#">Remove nth node from end of list</a>		
Linked List	<a href="#">Reorder List</a>		
Linked List	<a href="#">Detect and remove loop in a linked list</a>		
Linked List	<a href="#">Write a Function to get the Intersection Point of two Linked Lists</a>		
Linked List	<a href="#">Flatten a linked list with next and child pointers</a>		
Linked List	<a href="#">Linked list in zig-zag fashion</a>		
Linked List	<a href="#">Reverse a doubly linked list</a>		
Linked List	<a href="#">Delete nodes which have a greater value on right side</a>		
Linked List	<a href="#">Segregate even and odd Elements in a Linked List</a>		
Linked List	<a href="#">Point to next higher value node in a linked list with an Arbitrary Pointer</a>		
Linked List	<a href="#">Rearrange a given linked list in place</a>		
Linked List	<a href="#">Sort Biotonic Doubly Linked Lists</a>		
Linked List	<a href="#">Merge K Sorted Lists</a>		
Linked List	<a href="#">Merge sort for linked list</a>		Important
Linked List	<a href="#">Quicksort on singly-linked list</a>		Important
Linked List	<a href="#">Sum of two linked lists</a>		
Linked List	<a href="#">Flattening a linked list</a>		
Linked List	<a href="#">Clone a linked list with next and random Pointer</a>		
Linked List	<a href="#">Subtract two numbers represented as linked lists</a>		
Stacks & Queues	<a href="#">Implement two stacks in an Array</a>		
Stacks & Queues	<a href="#">Evaluation of Postfix Expression</a>		
Stacks & Queues	<a href="#">Implement Stack using Queues</a>		
Stacks & Queues	<a href="#">Queue Reversal</a>		
Stacks & Queues	<a href="#">Implement Stack Queue using Deque</a>		
Stacks & Queues	<a href="#">Reverse first k elements of queue</a>		
Stacks & Queues	<a href="#">Design Stack with Middle Operation</a>		
Stacks & Queues	<a href="#">Infix to Postfix</a>		
Stacks & Queues	<a href="#">Design and Implement Special stack</a>		
Stacks & Queues	<a href="#">Longest Valid String</a>		
Stacks & Queues	<a href="#">Find if an expression has duplicate parenthesis or not</a>		
Stacks & Queues	<a href="#">Stack permutations check if an array is stack permutation of other</a>		
Stacks & Queues	<a href="#">Count natural numbers whose permutation greater number</a>		
Stacks & Queues	<a href="#">Sort a stack using Recursion</a>		
Stacks & Queues	<a href="#">Queue based approach for first non repeating character in a stream</a>		
Stacks & Queues	<a href="#">The Celebrity Problem</a>		
Stacks & Queues	<a href="#">Next larger Element</a>		
Stacks & Queues	<a href="#">Distance of nearest cell</a>		
Stacks & Queues	<a href="#">Rotten-oranges</a>		
Stacks & Queues	<a href="#">Next smaller element</a>		
Stacks & Queues	<a href="#">Circular-tour</a>		
Stacks & Queues	<a href="#">Efficiently implement k-stacks single array</a>		
Stacks & Queues	<a href="#">The celebrity problem</a>		
Stacks & Queues	<a href="#">Iterative tower of hanoi</a>		
Stacks & Queues	<a href="#">Find the maximum of minimums for every window size in a given array</a>		

Stacks & Queues	<a href="#">lru cache implementation</a>		
Stacks & Queues	<a href="#">Find a tour that visits all stations</a>		
Greedy	<a href="#">Activity selection problem greedy algo</a>		
Greedy	<a href="#">Greedy algorithm to find minimum number of coins</a>		
Greedy	<a href="#">Minimum sum two numbers formed digits array-2</a>		
Greedy	<a href="#">Minimum sum absolute difference pairs two arrays</a>		
Greedy	<a href="#">Find maximum height pyramid from the given array of objects</a>		
Greedy	<a href="#">Minimum cost for acquiring all coins with k extra coins allowed with every coin</a>		
Greedy	<a href="#">Find maximum equal sum of every three stacks</a>		
Greedy	<a href="#">Job sequencing problem</a>		
Greedy	<a href="#">Greedy algorithm egyptian fraction</a>		
Greedy	<a href="#">Fractional knapsack problem</a>		
Greedy	<a href="#">Maximum length chain of pairs</a>		
Greedy	<a href="#">Find smallest number with given number of digits and digit sum</a>		
Greedy	<a href="#">Maximize sum of consecutive differences circular-array</a>		
Greedy	<a href="#">paper-cut minimum number squares</a>		
Greedy	<a href="#">Lexicographically smallest array-k consecutive swaps</a>		
Greedy	<a href="#">Problems-CHOCOLA</a>		
Greedy	<a href="#">Find minimum time to finish all jobs with given constraints</a>		
Greedy	<a href="#">Job sequencing using disjoint set union</a>		
Greedy	<a href="#">Rearrange characters string such that no two adjacent are same</a>		
Greedy	<a href="#">Minimum edges to reverse to make path from a source to a destination</a>		
Greedy	<a href="#">Minimize Cash Flow among a given set of friends who have borrowed money from each other</a>		
Greedy	<a href="#">Minimum Cost to cut a board into squares</a>		
Binary Trees	<a href="#">Maximum Depth of Binary Tree</a>		
Binary Trees	<a href="#">Reverse Level Order Traversal</a>		
Binary Trees	<a href="#">Subtree of Another Tree</a>		
Binary Trees	<a href="#">Invert Binary Tree</a>		
Binary Trees	<a href="#">Binary Tree Level Order Traversal</a>		
Binary Trees	<a href="#">Left View of Binary Tree</a>		
Binary Trees	<a href="#">Right View of Binary Tree</a>		
Binary Trees	<a href="#">ZigZag Tree Traversal</a>		
Binary Trees	<a href="#">Create a mirror tree from the given binary tree</a>		
Binary Trees	<a href="#">Leaf at same level</a>		
Binary Trees	<a href="#">Check for Balanced Tree</a>		
Binary Trees	<a href="#">Transform to Sum Tree</a>		
Binary Trees	<a href="#">Check if Tree is Isomorphic</a>		
Binary Trees	<a href="#">Same Tree</a>		
Binary Trees	<a href="#">Construct Binary Tree from Preorder and Inorder Traversal</a>		
Binary Trees	<a href="#">Height of Binary Tree</a>		
Binary Trees	<a href="#">Diameter of a Binary Tree</a>		
Binary Trees	<a href="#">Top View of Binary Tree</a>		
Binary Trees	<a href="#">Bottom View of Binary Tree</a>		
Binary Trees	<a href="#">Diagonal Traversal of Binary Tree</a>		
Binary Trees	<a href="#">Boundary Traversal of binary tree</a>		
Binary Trees	<a href="#">Construct Binary Tree from String with Brackets</a>		
Binary Trees	<a href="#">Minimum swap required to convert binary tree to binary search tree</a>		
Binary Trees	<a href="#">Duplicate subtree in Binary Tree</a>		
Binary Trees	<a href="#">Check if a given graph is tree or not</a>		
Binary Trees	<a href="#">Lowest Common Ancestor in a Binary Tree</a>		
Binary Trees	<a href="#">Min distance between two given nodes of a Binary Tree</a>		
Binary Trees	<a href="#">Duplicate Subtrees</a>		
Binary Trees	<a href="#">Kth ancestor of a node in binary tree</a>		
Binary Trees	<a href="#">Binary Tree Maximum Path Sum</a>		
Binary Trees	<a href="#">Serialize and Deserialize Binary Tree</a>		
Binary Trees	<a href="#">Binary Tree to DLL</a>		
Binary Trees	<a href="#">Print all k-sum paths in a binary tree</a>		
Binary Search Trees	<a href="#">Lowest Common Ancestor of a Binary Search Tree</a>		
Binary Search Trees	<a href="#">Binary Search Tree   Set 1 (Search and Insertion)</a>		
Binary Search Trees	<a href="#">Minimum element in BST</a>		
Binary Search Trees	<a href="#">Predecessor and Successor</a>		
Binary Search Trees	<a href="#">Check whether BST contains Dead End</a>		
Binary Search Trees	<a href="#">Binary Tree to BST</a>		
Binary Search Trees	<a href="#">Kth largest element in BST</a>		
Binary Search Trees	<a href="#">Validate Binary Search Tree</a>		
Binary Search Trees	<a href="#">Kth Smallest Element in a BST</a>		
Binary Search Trees	<a href="#">Delete Node in a BST</a>		
Binary Search Trees	<a href="#">Flatten BST to sorted list</a>		
Binary Search Trees	<a href="#">Preorder to Postorder</a>		
Binary Search Trees	<a href="#">Count BST nodes that lie in a given range</a>		
Binary Search Trees	<a href="#">Populate Inorder Successor for all Nodes</a>		
Binary Search Trees	<a href="#">Convert Normal BST to Balanced BST</a>		
Binary Search Trees	<a href="#">Merge two BSTs</a>		
Binary Search Trees	<a href="#">Given n appointments, find all conflicting appointments</a>		
Binary Search Trees	<a href="#">Replace every element</a>		
Binary Search Trees	<a href="#">Construct BST from given preorder traversal</a>		
Binary Search Trees	<a href="#">Find median of BST in O(n) time and O(1) space</a>		
Binary Search Trees	<a href="#">Largest BST in a Binary Tree</a>		Important

Heaps & Hashing	<a href="#">Choose k array elements such that difference of maximum and minimum is minimized</a>		
Heaps & Hashing	<a href="#">Heap Sort</a>		
Heaps & Hashing	<a href="#">Top K Frequent Elements</a>		
Heaps & Hashing	<a href="#">k largest elements in an array</a>		
Heaps & Hashing	<a href="#">Next Greater Element</a>		
Heaps & Hashing	<a href="#">K'th Smallest/Largest Element in Unsorted Array</a>		
Heaps & Hashing	<a href="#">Find the maximum repeating number in O(n) time and O(1) extra space</a>		
Heaps & Hashing	<a href="#">K-th smallest element after removing some integers from natural numbers</a>		
Heaps & Hashing	<a href="#">Find k closest elements to a given value</a>		
Heaps & Hashing	<a href="#">K'th largest element in a stream</a>		
Heaps & Hashing	<a href="#">Connect Ropes</a>		
Heaps & Hashing	<a href="#">Cuckoo Hashing</a>		
Heaps & Hashing	<a href="#">Itinerary from a List of Tickets</a>		
Heaps & Hashing	<a href="#">Largest Subarray with 0 Sum</a>		
Heaps & Hashing	<a href="#">Count distinct elements in every window of size_k</a>		
Heaps & Hashing	<a href="#">Group Shifted Strings</a>		
Heaps & Hashing	<a href="#">Merge K Sorted lists</a>		
Heaps & Hashing	<a href="#">Find Median from Data Stream</a>		
Heaps & Hashing	<a href="#">Sliding Window Maximum</a>		
Heaps & Hashing	<a href="#">Find the smallest positive number</a>		
Heaps & Hashing	<a href="#">Find Surpasser Count of each element in array</a>		
Heaps & Hashing	<a href="#">Tournament Tree and Binary Heap</a>		
Heaps & Hashing	<a href="#">Check for palindrome</a>		
Heaps & Hashing	<a href="#">Length of the largest subarray with contiguous elements</a>		
Heaps & Hashing	<a href="#">Palindrome Substring Queries</a>		
Heaps & Hashing	<a href="#">Subarray distinct elements</a>		
Heaps & Hashing	<a href="#">Find the recurring function</a>		
Heaps & Hashing	<a href="#">K maximum sum combinations from two arrays</a>		
Graphs	<a href="#">BFS</a>		
Graphs	<a href="#">DFS</a>		
Graphs	<a href="#">Flood Fill Algorithm</a>		
Graphs	<a href="#">Number of Triangles</a>		
Graphs	<a href="#">Detect cycle in a graph</a>		
Graphs	<a href="#">Detect cycle in an undirected graph</a>		
Graphs	<a href="#">Rat in a Maze Problem</a>		
Graphs	<a href="#">Steps by Knight</a>		
Graphs	<a href="#">Clone graph</a>		
Graphs	<a href="#">Number of Operations to Make Network Connected</a>		
Graphs	<a href="#">Dijkstra's shortest path algorithm</a>		
Graphs	<a href="#">Topological Sort</a>		
Graphs	<a href="#">Oliver and the Game</a>		
Graphs	<a href="#">Minimum time taken by each job to be completed given by a Directed Acyclic Graph</a>		
Graphs	<a href="#">Find whether it is possible to finish all tasks or not from given dependencies</a>		
Graphs	<a href="#">Find the number of islands</a>		
Graphs	<a href="#">Prim's Algo</a>		
Graphs	<a href="#">Negative Weighted Cycle</a>		
Graphs	<a href="#">Floyd Warshall</a>		
Graphs	<a href="#">Graph Coloring</a>		
Graphs	<a href="#">Snakes and Ladders</a>		
Graphs	<a href="#">Kosaraju's Theorem</a>		
Graphs	<a href="#">Journey to moon</a>		
Graphs	<a href="#">Vertex Cover</a>		
Graphs	<a href="#">M Coloring Problem</a>		
Graphs	<a href="#">Cheapest Flights Within K Stops</a>		
Graphs	<a href="#">Find if there is a path of more than k length from a source</a>		
Graphs	<a href="#">Bellman Ford</a>		
Graphs	<a href="#">Bipartitie Graph</a>		
Graphs	<a href="#">Word-Ladder</a>		
Graphs	<a href="#">Allen Dictionary</a>		
Graphs	<a href="#">Kruskals MST</a>		Important
Graphs	<a href="#">Total number spanning trees graph</a>		
Graphs	<a href="#">Travelling Salesman</a>		Important
Graphs	<a href="#">Find longest path directed acyclic graph</a>		
Graphs	<a href="#">Two Clique Problem</a>		
Graphs	<a href="#">Minimise the cash flow</a>		
Graphs	<a href="#">Chinese postman</a>		
Graphs	<a href="#">Water Jug</a>		
Graphs	<a href="#">Water Jug 2</a>		
Tries	<a href="#">Construct a trie from scratch</a>		
Tries	<a href="#">Print unique rows in a given boolean matrix</a>		
Tries	<a href="#">Word Break Problem   (Trie solution)</a>		
Tries	<a href="#">Given a sequence of words, print all anagrams together</a>		
Tries	<a href="#">Find shortest unique prefix for every word in a given list</a>		
Tries	<a href="#">Implement a Phone Directory</a>		
DP	<a href="#">Knapsack with Duplicate Items</a>		
DP	<a href="#">BBT counter</a>		
DP	<a href="#">Reach a given score</a>		
DP	<a href="#">Maximum difference of zeros and ones in binary string</a>		
DP	<a href="#">Climbing Stairs</a>		
DP	<a href="#">Permutation Coefficient</a>		

DP	<a href="#">Longest Repeating Subsequence</a>		
DP	<a href="#">Pairs with specific difference</a>		
DP	<a href="#">Longest subsequence-1</a>		
DP	<a href="#">Coin Change</a>		
DP	<a href="#">LIS</a>		
DP	<a href="#">Longest Common Subsequence</a>		
DP	<a href="#">Word Break</a>		
DP	<a href="#">Combination Sum IV</a>		
DP	<a href="#">House Robber</a>		
DP	<a href="#">Houe Robber 2</a>		
DP	<a href="#">Decode Ways</a>		
DP	<a href="#">Unique Paths</a>		
DP	<a href="#">Jumps Game</a>		
DP	<a href="#">Knapsack Problem</a>		
DP	<a href="#">nCr</a>		
DP	<a href="#">Catalan Number</a>		
DP	<a href="#">Edit Distance</a>		
DP	<a href="#">Subset Sum</a>		
DP	<a href="#">Gold mine</a>		
DP	<a href="#">Assembly Line Scheduling</a>		
DP	<a href="#">Maximize The Cut Segments</a>		
DP	<a href="#">Maximum sum increasing subsequence</a>		
DP	<a href="#">Count all subsequences having product less than K</a>		
DP	<a href="#">Maximum sum increasing subsequence</a>		
DP	<a href="#">Egg dropping puzzle</a>		
DP	<a href="#">Max length chain</a>		
DP	<a href="#">Largest Square in Matrix</a>		
DP	<a href="#">Maximum Path Sum</a>		
DP	<a href="#">Minimum Number of Jumps</a>		
DP	<a href="#">Minimum removals from array to make max – min &lt;= K</a>		
DP	<a href="#">Longest Common Substring</a>		
DP	<a href="#">Partition Equal Subset Sum</a>		
DP	<a href="#">Longest Palindromic Subsequence</a>		
DP	<a href="#">Count Palindromic Subsequences</a>		
DP	<a href="#">Longest Palindromic Substring</a>		
DP	<a href="#">Longest Alternating Sequence</a>		
DP	<a href="#">Weighted Job Scheduling</a>		
DP	<a href="#">Coin Game</a>		
DP	<a href="#">Coin Game Winner</a>		
DP	<a href="#">Optimal Strategy for a game</a>		
DP	<a href="#">Word Wrap</a>		
DP	<a href="#">Mobile numeric keypad</a>		
DP	<a href="#">Maximum Length of Pair Chain</a>		
DP	<a href="#">Matrix Chain Multiplication</a>		
DP	<a href="#">Maximum profit by buying and selling a share at most twice</a>		
DP	<a href="#">Optimal BST</a>		
DP	<a href="#">Largest Submatrix with sum 0</a>		
DP	<a href="#">Largest area rectangular sub-matrix with equal number of 1's and 0's</a>		
Bit Manipulation	<a href="#">Count set bits in an integer</a>		
Bit Manipulation	<a href="#">Find the two non-repeating elements in an array of repeating elements</a>		
Bit Manipulation	<a href="#">Program to find whether a no is power of two</a>		
Bit Manipulation	<a href="#">Find position of the only set bit</a>		
Bit Manipulation	<a href="#">Count number of bits to be flipped to convert A to B</a>		
Bit Manipulation	<a href="#">Count total set bits in all numbers from 1 to n</a>		
Bit Manipulation	<a href="#">Copy set bits in a range</a>		
Bit Manipulation	<a href="#">Calculate square of a number without using *, / and pow()</a>		
Bit Manipulation	<a href="#">Divide two integers without using multiplication, division and mod operator</a>		
Bit Manipulation	<a href="#">Power Set</a>		
Segment Trees	<a href="#">Range Sum Query - Immutable</a>		
Segment Trees	<a href="#">Range Minimum Query</a>	Google Interview Qs	
Segment Trees	<a href="#">Range Sum Query - Mutable</a>		
Segment Trees	<a href="#">Create Sorted Array through Instructions</a>		
Segment Trees	<a href="#">Count of Range Sum</a>		
Segment Trees	<a href="#">Count of Smaller Numbers After Self</a>		