

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	
Hard	Ideal Time : 40-60 mins (based on Qs)	
Topics	Question	Remarks
Strings	<a href="#">Edit Distance</a>	use Dynaming Programming (if possible)
Searching & Sorting	<a href="#">Sort a Nearly Sorted (or K sorted) Array</a>	
Searching & Sorting	<a href="#">How to Efficiently Sort a Big List Dates in 20's</a>	
Searching & Sorting	<a href="#">find a repeating and a missing number</a>	
Searching & Sorting	<a href="#">sort array according count set bits</a>	
Searching & Sorting	<a href="#">Minimum Swaps to Make Two Array Identical</a>	
Searching & Sorting	<a href="#">Insert in Sorted and Non-Overlapping Interval Array</a>	
Searching & Sorting	<a href="#">3-Way QuickSort</a>	
Backtracking	<a href="#">Find if There is a Path of More Than k Length From a Source</a>	
Backtracking	<a href="#">Match a Pattern and String without Using Regular Expressions</a>	
Linked List	<a href="#">Josephus Circle implementation using STL list</a>	
Linked List	<a href="#">Find a triplet from three linked lists with sum equal to a given Number</a>	
Linked List	<a href="#">Pair with given sum</a>	
Linked List	<a href="#">Select a random node from a singly linked list</a>	
Linked List	<a href="#">First non repeating character</a>	
Stacks & Queues	<a href="#">Implement Stack using Queue or heap</a>	
Stacks & Queues	<a href="#">Sum of minimum-maximum elements subarrays size-k</a>	
Stacks & Queues	<a href="#">Minimum time required so that all oranges become rotten</a>	
Stacks & Queues	<a href="#">Efficiently implement k-queues single array</a>	

Greedy	<a href="#">Maximize array sum after k-negation operations</a>	
Greedy	<a href="#">Program for shortest job first or sjf-cpu scheduling set 1 non-preemptive</a>	
Binary Trees	<a href="#">Check Mirror in N-ary tree</a>	
Binary Trees	<a href="#">Maximum sum of nodes in Binary tree such that no two are adjacent</a>	
Binary Search Trees	<a href="#">Brothers From Different Roots</a>	
Heaps & Hashing	<a href="#">Check the condition</a>	
Heaps & Hashing	<a href="#">Check if an array can be divided into pairs whose sum is divisible by k</a>	
Heaps & Hashing	<a href="#">Design a effective DSA</a>	
Heaps & Hashing	<a href="#">Find number of Employees Under every Manager</a>	
Heaps & Hashing	<a href="#">Pancake Sorting</a>	
Graphs	<a href="#">Bride in a graph</a>	
Graphs	<a href="#">Seven Bridges of Königsberg</a>	
Graphs	<a href="#">Minimum edges to reverse to make path from a source to a destination</a>	
DP	<a href="#">Maximum Sum Rectangle</a>	
DP	<a href="#">Interleaved Strings</a>	
DP	<a href="#">Painting the Fence</a>	
DP	<a href="#">Largest independent Set</a>	
DP	<a href="#">Minimum cost to fill given weight in a bag</a>	
DP	<a href="#">Boolean Parenthesization</a>	
DP	<a href="#">Maximum Profit</a>	
DP	<a href="#">Palindromic Partitioning</a>	