CodeBix

Basic to Advanced DSA

Live class syllabus

Website :- codebix.in/

Whatsaap :- +91 8700352711

Youtube :- https://www.youtube.com/@codebix1096

Month 1: Fundations and Linear Data Structures

TOPICS - Array, String, Linked List, Basic Recursion, Searching algo , Sorting algo, Stack and Queues

Week 1: Introduction to Algorithms and Basics

- Theory
    - What are DATA STRUCTURES AND ALGORITHMS ?
    - Understand Arrays, Strings and List
    - Understand basic Recursion
    - Understand time and space complexity
- Problems :
    - Implement ArrayList - Data Structure
    - Two Sum
    - Move Zeroes
    - Best Time to Buy and Sell Stock
    - Find the Duplicate Number
    - Reverse String
    - Rotate Matrix
    - Sort Colors
    - Valid Anagram
    - Longest Substring Without Repeating Characters
    - Find Factorial using recursion
    - Find Fibonacci no

Week 2: **Searching and Sorting**

- Understand Searching Algorithms
- Understand Searching Algorithms
- Problems:
    - Implement Linear search
    - Implement Binary search - Both recursive and iterative
    - Search in Rotated Sorted Array
    - First Bad Version
    - Search a 2D Matrix
    - Find First and Last Position of Element in Sorted Array
    - Find Minimum in Rotated Sorted Array
    - Number of occurrence - sorted array find number of occ of a no.
    - Search Insert Position
    - Find Ceil and Floor
    - Find Minimum in Rotated Sorted Array II
    - Find Peak Element
    - Peak Index in a Mountain Array
    - Pow(x, n)
    - Find position of an element in a sorted array of infinite numbers
    - Find Smallest Letter Greater Than Target
    - Search a 2D Matrix
    - Implement Bubble sort
    - Implement Insertion sort
    - Implement Selection sort
    - Implement Merge and QuickSort

Week 3: Linked Lists

- Understand what are singly, doubly and circular linked lists?
- Learn basics like what is head, tail, nodes and pointers?
- Problems :
    - Count no of nodes in LinkedList
    - Insert node in LinkedList
    - Delete node in LinkedList with given head
    - Delete Node in a Linked List without head
    - Remove Duplicates from Sorted List
    - Reverse Linked List
    - Find middle of LinkedList
    - Linked List Cycle
    - Find Nth node from End
    - Linked List Components
    - Remove Nth node from End
    - Remove Duplicates from Sorted List
    - Remove Nodes From Linked List which are greater in right
    - Merge Two Sorted Lists Reverse Linked List
    - Detect Cycle in a Linked List
    - Intersection of Two Linked Lists
    - Palindrome Linked List
    - Copy List with Random Pointer
    - Add Two Numbers II
    - Partition List

Week 4: **Stacks and Queues**

- Understand what Stacks and Queues?
- Problems :
    - Implement Stack and Queue using Array
    - Implement Stack and Queue using LinkedList
    - Implement Queue using Stacks

    Bracket pattern problems

    - Valid Parentheses
    - Minimum Add to Make Parentheses Valid
    - Min Stack

    NGE pattern problems

    - Next Greater Element right
    - Next Smaller Element left
    - Next Greater Element right
    - Next Smaller Element left
    - Daily Temperatures
    - Online Stock Span
    - Largest Rectangle in Histogram

Other

- Decode
- Design Circular Queue
- Evaluate Reverse Polish Notation
- Merge Intervals
- Trapping Rain Water
- Asteroid Collision

Month 2: Non Linear Data Structures

TOPICS - N ary Tree, Binary Tree, Binary Search Tree and Graphs

Week 5: Tree - 1 - N ary tree

- Understand N - ary tree data structure
- What are root, leaf, Tree Nodes, children, pointers and branches etc?
- Problems:
    - Implement N ary tree

Traversal of N ary tree

- Preorder Traversal on N ary tree
- Postorder Traversal on N ary tree

- Level order Traversal on N ary Tree
- ZigZag traversal

DFS or BFS N ary tree

- Count number of nodes in N ary tree
- Count leafs of n ary tree
- Maximum Depth of N-ary Tree
- Node With Data

Only DFS N ary tree

- Maximum Depth of N-ary Tree
- Lowest common ancestor in N ary tree

Only BFS N ary tree

- Cousins in N ary

Week 6: Tree - 2 - Binary tree

- Understand Binary tree data structure
- What are root, leaf, Tree Nodes, children, pointers and branches etc?
- Problems :
    - Implement Binary tree

    Traversal of Binary tree

    - Preorder traversal on Binary Tree
    - Postorder traversal on Binary Tree
    - Inorder traversal on Binary Tree
    - Level order traversal on Binary Tree
    - ZigZag traversal
    - Boundary Traversal

    Only DFS - Binary Tree

    - Maximum Depth of Binary Tree
    - Minimum Depth of Binary Tree
    - Sum Root to Leaf Numbers
    - Binary Tree Tilt
    - Binary Tree Paths
    - Diameter of Binary Tree
    - Lowest Common Ancestor of a Binary Tree

- Recover a Tree From Preorder Traversal
- Path Sum II
- Path Sum III
- Min distance between two given nodes of a Binary Tree

Only BFS - Binary Tree

- Cousins in Binary Tree
- Vertical Order Traversal of a Binary Tree
- All Nodes Distance K in Binary Tree

Both DFS and BFS - Binary Tree

- Same Tree
- Path Sum
- Invert Binary Tree
- Sum of Left Leaves
- Merge Two Binary Trees
- Find Bottom Left Tree Value
- Populating Next Right Pointers in Each Node

Week 7: Tree - 3 - Binary tree cont. And Binary Search Tree

- Understand Binary Search tree data structure
- Problems:

Tree views Pattern - Try with Only BFS

- Left view of Binary Tree
- Right view of Binary Tree
- Top view of Binary Tree
- Bottom view of Binary Tree

Tree Construction Pattern - Try with only DFS

- Construct Binary Tree from Inorder and Postorder Traversal
- Construct Binary Tree from Preorder and Inorder Traversal
- Construct Full Binary Tree from given preorder and postorder traversals
- Construct Binary Search Tree from Preorder Traversal

Binary Search Tree

- Convert Sorted Array to Binary Search Tree
- Validate Binary Search Tree
- Inorder Successor in BST
- Search in a Binary Search Tree
- Lowest Common Ancestor of a Binary Search Tree
- Kth Smallest Element in a BST

- Trim a Binary Search Tree

Week 8: Graph - 1

- Understand graph data structure
- Directed and Undirected GraphsTraversal on Graph - BFS, DFS and Topological Sort
- Different ways to implement Graph and which is better in which condition?
- Problems :

Implementation and Basics

- Construct Graph using edge list
- Construct Graph using Adjacency List
- Construct Graph using Adjacency Matrix
- Construct Graph on Grids
- DFS on Adjacency List
- DFS on Adjacency Matrix
- DFS on Grid
- BFS on Adjacency List
- BFS on Adjacency Matrix
- FS on Grid

Graph Adjacency matrix and list - Both BFS and DFS

- Keys and Rooms
- Number of Provinces
- All Paths From Source to Target
- Number of Operations to Make Network Connected - also try with union find
- Is Graph Bipartite?
- Clone Graph
- Cycle in an Undirected Graph
- Cycle in an Directed Graph

Topological Sort

- Topological Sort using BFS and DFS
- Course Schedule
- Course Schedule II
- Alien dictionary
-

Week 9: Graph - 2

- Disjoint Set and problems on Graph on Grid
- Problems:

Union Find problems

- Number of Operations to Make Network Connected
- Find if Path Exists in Graph

Graph on Grids

- DFS on Grids

- Number of Islands - also try with BFS 🙂
- Number of Closed Islands
- Max Area of Island
- Surrounded Regions
- Island Perimeter
- Flood Fill
- Word Search
- Making A Large Island

- BFS on Grids

- Rotting Oranges
- 01 Matrix
- Snakes and Ladders
- Minimum steps to reach target by a Knight

Month 3: Advanced Data Structures

TOPICS - HashMaps, Heaps, Priority Queue, Bit Manipulation, Advanced Recursion and BackTracking

Week 10: Maps and Tries

- Understand different types of Maps like - HashMap, TreeMap and LinkedMap
- Trie Data Structure
- Problems:

Map

- Implement HashMap
- Group Anagrams
- Isomorphic Strings
- Roman to Integer
- Contiguous Array
- Valid Anagram
- Letter Combinations of a Phone Number
- Subarray Sum Equals K
- Dot Product Of Two Sparse Vectors
- Isomorphic Strings

Trie

- Implement Trie Data Structure
- Lexicographical Numbers
- Word Break
- Word Break II

Week 11: Heap, Priority Queue and Advanced Graph topics

- Understand what is heap, minPriority queue, maxPriority Queue and heapSort etc
- Learn Advanced graph topics majorly for weighted graphs Minimum Spanning Tree and important algorithms like Dijkstra's algorithm and Prim's algorithm etc
- Problems :

Heap and Priority Queue

- Implement minPriority Queue and maxPriority Queue
- Kth Largest Element in an Array
- Merge k Sorted Lists
- Find Median from Data Stream
- Reorganize String
-

Advanced Graph topics - weighted Graph

- Implement Dijkstra's algorithm on Adjacency matrix and List
- Cheapest Flights Within K Stops

- Reachable Nodes In Subdivided Graph
- Implement Prim's algorithm on Adjacency matrix and List
- Remove Max Number of Edges to Keep Graph Fully Traversable

Week 12: Bit Manipulation and Sliding Windows

- Understand topics like Binary Representation on no, Bitwise Operators, Bitwise Manipulation Techniques, Hamming Codes and kernighan's Algorithm etc
- Understand Sliding window technique
- Problems :

Bit Manipulation

- Setting a particular bit.
- Clearing a particular bit.
- Flipping a particular bit.
- Checking if a bit is set or not.
- Counting set bits
- Check if a number is even or odd
- Convert to Base -2
- Create RSB mask
- Detect if two integers have opposite signs
- Swap number without any arithmetic operator
- Bitwise AND of Numbers Range
- Find the Duplicate Number

- Power of Two
- Missing Number
- Single Number
- Single Number II
- Single Number III

Sliding window

- Minimum Size Subarray Sum
- Maximum Sum Subarray of Size K
- Longest Substring Without Repeating Characters
- Trapping Rain Water
- Find All Anagrams in a String
- Permutation in String
- Longest Substring with At Most K Distinct Characters

Week 13: Backtracking and Advanced Recursion

- Understand Advanced Recursion and Backtracking technique
- Problems:
  - Subsets
  - Subsets II
  - Permutations

- Permutations II
- Combination Sum
- Combination Sum II
- Word Search
- Letter Combinations of a Phone Number
- Palindrome Partitioning
- Generate Parentheses
- N-Queens
- N-Queens II
- Sudoku Solver
- Word Ladder II
- Knight Probability in Chessboard

Month 4: Dynamic Programming and Greedy

Week 14: Dynamic Programing - 1

- Understand Dynamic Programing and ways to use it
- Top-down approach (Memoization)
- Bottom-up approach (Tabulation)
- Understand different patterns of Dynamic programing
- Problems:

0/1 Knapsack

- 0/1 Knapsack Problem
- Partition Equal Subset Sum

- Subset Sum Problem
- Minimum Subset Sum Difference
- Count of subset sum
- Target Sum (Leetcode)

Unbounded Knapsack

- Unbounded Knapsack (Repetition of items allowed)
- Minimum Cost to Cut a Stick
- Coin Change
- Minimum Coin Change
- Maximum Ribbon Cut

Week 15: Dynamic Programing - 2

- Problems :

Fibonacci Numbers

- Fibonacci Numbers
- Climbing Stairs
- Number divisors
- Minimum jumps to reach end
- House Robber
- House Robber II

Stocks problems

- Best Time to Buy and Sell Stock

- Best Time to Buy and Sell Stock II
- Best Time to Buy and Sell Stock with Cooldown
- Best Time to Buy and Sell Stock with Transaction Fee
- Best Time to Buy and Sell Stock IV
- Maximum Score Words Formed by Letters

Palindromic Subsequence

- Longest Pallindromic Subsequence
- Longest Pallindromic Substring
- Count of Pallindromic Substrings
- Minimum deletions to make a string pallindrome

Week 16: Dynamic Programing - 3 and Greedy

- Understand Greedy approach and when it is used?
- Problems:

Longest Common Substring

- Longest Common Substring
- Longest Common Subsequence
- Minimum Deletions and Insertions to Transform a String into another
- Longest Increasing Subsequence
- Maximum Sum Increasing Subsequence
- Shortest Common Supersequence

- Minimum deletions to make sequence sorted
- Longest repeating subsequence
- Longest Bitonic Subsequence
- Longest Alternating Subsequence
- Edit Distance
- String Interleaving

Greedy

- Distribute Candies
- Gas Station
- Maximum Subarray
- Queue Reconstruction by Height
- Candy
- Non-overlapping Intervals
- Increasing Triplet Subsequence