Competitive Programming Roadmap

- Use A2OJ Ladders to practice upon your skills
- Alternate: Practice one day, Study next day
- Sites for practicing topics from tags: Codeforces, SPOJ and Hackerearth

Roadmap

- Learn Algorithm & Practice Problems on alternate days

Basic

- Pattern printing problems
- Analysis of time complexity
- Linear Search problems
- Circular array using simple array
- Palindrome, Perfect, Armstrong, Strong number
- Simple Hashing problems
- Prefix Sum Problems 1D/2D
- Sliding window technique (1/5 of Questions)

Intermediate

- Binary Search problems (2/5 of Questions)
- Find GCD of 2 numbers in O($\log n$)
  - (Euclidean and Extended Euclidean algorithm)
- Checking Primes in O($\sqrt{n}$)
- Sieve of Eratosthenes
- Segmented Sieve
- Finding the prime factorization of a number in O($\log n$) per query
- Euler Totient function
- Fermat Little theorem

Number Theory

- Finding x^n in O($\log n$)
- Modular Arithmetic
- Module Inverse of a number
- Chines Remainder Theorem
- Factorial Modulo Mod
- Finding ${}_nC^r$ & ${}_nP^r$ for queries in O(1)
- Inclusion Exclusion principle

Some Advanced

- Learn about basic sorting Algorithms (Bubble, Selection, Insertion)
- Constructive and having swap terms in it
- Bit Manipulation (Left Shift, Right Shift, Set bit, MSB, LSB.)
    - (Hackerearth has good Tutorials)
- Power Set of a given array or string using BIT.
- Number of subarrays with XOR as ZERO
    - (Not an algorithm, but a must do problem)
-
- Solve Problems related to Greedy Algorithms
- Kadane's Algorithms and problem related to them
- Job sequence and activity selection problem

Recursion

- Implement basic problems like finding Factorial
- Implement Binary search using Recursion
- Implement Modular Exponentiation
- Solve recursion problem like finding subset with given sum and other problems

Advanced

- Learn Merge Sort & Quick sort algorithms (Problem: Count Inversion)
- Do backtracking problems like Sudoku and N-Queen problem
  - (Will help in Dynamic Programming problems)
- Meet in the Middle Algorithm and problems
- Solve Problems for Divide & Conquer problems on Codeforces
- Find Next Greater/Smaller element using Stack
- Solve Problems related to Parenthesis Problem
- Largest Rectangular Area in Histogram
- Problem related to Heap (Use PriorityQueue in STL)

More Advanced Problems

- Hashing on strings, understand when collision happens
  - (https://cp-algorithms.com/)
- Rabin Karp algorithm
- Prefix function
- KMP Algo
- Z-Function
- Manacher's Algorithm

Trees

- Tree/Graph representation
- DFS/BFS traversal in tree /graph
- Diameter of a tree/Height/
- Euler Tour of tree
- Finding LCA using Euler Tour/Binary Lifting
- Distance between two nodes
- Solve Subtree Problems

Graph

- Connected Components
- Topological sort
- Cyclic detection in graph
- Bipartite check in graph
- SCC using Kosaraju's algorithm
- Dijkstra's Algo
- Bellman Ford Algo
- Floyd Warshall algorithm
- Bridge in Graph
- Articulation point in graph
- Minimum spanning tree & Kruskal algo
- Prim's Algorithm
- 0/1 BFS in linear time

Dynamic Programming

- Start with Recursion & Memorization with great knowledge
- Knapsack prob solve
- Solve All Problems of AtCoder’s Educational contest
- Solve problem from SPOJ then Codeforces
- Understand how we write recurrence for Digit DP (CF blog)
- Read about Dynamic Programming with bitmasks
- Dynamic Programming in trees
- SOS Dynamic Programming

More

- 1: Disjoint Set (Using all optimizations)
- 2: Offline Queries using Disjoint Set
- 3: Kruskal's Algorithm using Disjoint Set
- 4: Sparse Table
- 5: Fenwick Tree (Read Update Trick also)
- 6: Binary Lifting on Fenwick tree (More Solve prob)

- Matrix Exponentiation
- Sqrt Decomposition Technique
- Update and query operations
- Mo's Algo
- Mo's Algo on Trees
- Segment Tree
    - (Range queries and point updates)
- Lazy propagation in segment tress
- Sprague-Grundy Theorem
- Flows and Related problem
- Heavy light decomposition
- Convex Hull Algorithm
- FFT/NTT