

## Time Complexity :

### Question Types:

- 1.Basic Primer [4]
- 2.Function Calling itself [3]
- 3.Math [4]
- 4.Amortized Complexity [1]
- 5.Compare Function [3]

## Arrays :

- 1.Primers [3]
- 2.Examples [3]
- 3.Array Math [10]
- 4.Hash Search [1]
- 5.Sorting [7]
- 6.Value Ranges [3]
- 7.Space Recycle [3]
- 8.Simulation Array [9]
- 9.Missing/Repeated number [3]
- 10.Bucketing [4]
- 11.Arrangement [5]

## Mathmatics:

- 1.Primers [2]
- 2.Examples [5]
- 3.Adhoc [8]
- 4.Hash Search [1]
- 5.Bit Play [1]
- 6.Base Conversion [2]
- 7.Number Encoding [2]
- 8.Digital Operation [3]
- 9.Number Theory [8]
- 10.Combinatories [3]

## Binary Search :

- 1.Basic Example [2]
- 2.Simple Binary Search [6]
- 3.Search Step Simulation [2]
- 4.Search Answer [6]
- 5.Sort Modification [2]

## Strings :

- 1.String Simulation [10]
- 2.String Parsing [4]
- 3.String Search [3]
- 4.Pretty Print [3]
- 5.String Tricks [5]
- 6.String Math [6]
7. Words [2]

## Bit Manipulation :

- 1.Bit Play [5]
- 2.Bit Array [2]
- 3.Bit Tricks [4]

## Two Pointers

- 1.Sorting [4]
- 2.Multipile Arrays [2]
- 3.Tricks [6]
- 4.Inplace Update [4]

## Linked-List

- 1.Examples [2]
- 2.List Sort [4]
- 3.List Trick [4]
- 4.List 2 Pointer [5]
- 5.List Math [2]
- 6.Pointer Move [4]

## Stack and Queues

- 1.Simple Examples [2]
- 2.Stack Simple [4]
- 3.Queue [2]
- 4.CleverStack [3]
- 5.Stack Math [2]
- 6.BFS [1]

## BackTracking :

- 1.Simple Examples [2]
- 2.Subsets [6]
- 3.Permutations [1]
- 4.Game Solving [2]
- 5.Maths and Backtracking [3]
- 6.Bruteforce Builder [3]

## Hashing:

- 1.Hash Search [8]
- 2.Incremental Hash [4]
- 3.Hashing Two Pointer [3]
- 4.Key Formation [4]
- 5.Maths and Hashing [3]

## Heap and Maps :

- 1.Heap [9]
- 2.map [2]

## Tree DataStructure :

- 1.Basic Examples [1]

- 2.Tree Construction [5]
- 3.Traversal [7]
- 4.BST traversal [5]
- 5.Level Order [5]
- 6.Simple Tree Operation [7]
- 7.Root to Leaf [6]
- 8.Inplace Change [1]
- 9.Two Trees [3]
- 10.Tree Search [1]
11. Linked Tree [1]
12. Segment Tree [1]

## Greedy Algorithms

- 1.Easy Greedy [4]
- 2.Medium Greedy [6]

## Dynamic Programming

- 1.2D string DP [9]
- 2.Simple Array DP [9]
- 3.Greedy OR DP [3]
- 4.Derived DP [3]
- 5.KnapSack [4]
- 6.DP [1]
- 7.Adhoc [1]
- 8.DP Optimized backtrack [1]
- 9.Multiply DP [2]
- 10.Breaking Words [2]
- 11.DP tricky [11]
- 12.Tree DP [2]
- 13.Matrix DP [14]
- 14.Suffix/Prefix DP [5]

## Graph :

- 1.Graph Traversal [5]
- 2.DFS [6]
- 3.BFS [8]
- 4.Graph Connectivity [6]
- 5.Graph Adhoc [1]
- 6.Shortest Path [5]
- 7.Graph Hashing [1]