**Week 1**

o **Day-1 :** Introduction : Asymptotic Analysis (Finding time and space complexities)

o **Day-2 :** Mathematics : GCD, Prime, Factorial, Sieve of Eratosthenes, Computing Power

o **Day-3, 4 :** Arrays : Types, Operations on Arrays, Sliding Window Technique and Prefix Sum

o **Day - 5, 6 :** Basic Recursion

**Week 2:**

o **Day-1, 2 :** Bit Magic : Bit Operators,Tricks to use bit

manipulation.

o **Day-3, 4** : Matrix :Operations on Matrix(Search, Delete, Insert, Rotate,Transpose,)

o **Day-5, 6 :** Searching: Linear Search, Binary Search, Two pointer approach..

**Week 3:**

o **Day-1, 2, 3 :** Sorting: Insertion Sort, Merge Sort, Quick Sort, Cycle Sort, Counting Sort, Radix Sort, Bucket Sort, Custom Sort using STL

o **Day-4, 5, 6 :** Hashing: Different Types of Hashing Techniques, Collision resolution Techniques, Hashing Questions

**Week 4:**

o **Day-1, 2, 3:**Strings: Basic Operations, Naive Pattern Search, Other searching algorithms(KMP, Rabin-Karp).

o **Day-4, 5, 6:**Linked Lists: Singly Linked List, Doubly Linked Lists, Circular Linked List, Skip List, Doubly Circular Linked List, Questions

**Week 5:**

o **Day-1, 2:** Stacks: Stack Operations, Implementation, Questions

o **Day-3, 4:** Queues: Queue Operations, Implementation, Different Questions.

o **Day-5, 6:** Deque Operations, Implementation, Different Questions.

**Week 6:**

o **Day-1, 2, 3 :** Tree: Binary Tree, Tree Traversals, Different Questions

o **Day-4, 5, 6 :** Binary Search Tree: Search, Insert, Delete and other important questions, AVL (Basic Introduction),Self

Balancing Trees and their use in sets and maps STL.

**Week 7:**

o **Day 1, 2:** Heaps: Binary Heap(Min and Max Heap), Priority Queue, Questions based on heaps.

o **Day-3, 4, 5, 6:**Graphs: Graph Implementation,Traversals, Cycle Detection, Bipartite Graph, Minimum Spanning Tree,

Topological Sorting.

**Week 8:**

o **Day 1, 2 :** Greedy : Fractional Knapsack, Activity Selection, Job Sequencing

o **Day 3, 4, 5, 6 :** Dynamic Programming: Properties (Top Down, Bottom Up, Optimal Substructures, Overlapping Subproblems) and Standard Problems (LIS, LCS, etc)

**Week 9:**

o Dynamic Programming Problems (Variations of Standard Problems)

o Graph Algorithms

▪ Shortest Path Algorithms

▪ Connected Components

▪ Bridges

**Week 10:**

o **Day 1, 2:** Tries

o **Day 3, 4:** Segment Tree

o **Day 5, 6:** Disjoint Set Union: Operations(Union, Find), Path Compression