1. **Arrays**
2. Find the element that appears once TC=(log n)
3. Longest/count [Subarray with given sum](https://takeuforward.org/data-structure/longest-subarray-with-given-sum-k/" \t "https://takeuforward.org/strivers-a2z-dsa-course/strivers-a2z-dsa-course-sheet-2/_blank) TC = O(n) SC = O(n)
4. Find all pairs with a given sum TC = O(n log n) SC = O(n)
5. Next Permutation Tc = O(n)
6. Longest Consecutive Sequence in an Array TC = O(n) SC = O(n)
7. Rotate matrix by 90 degree TC = O(n^2)
8. Spiral matrix TC = O(n\*m)
9. Majority 1/2 and 1/3 TC = O(n)
10. 3 sum
11. 4 sum
12. [Count number of subarrays with given xor K](https://takeuforward.org/data-structure/count-the-number-of-subarrays-with-given-xor-k/" \t "https://takeuforward.org/strivers-a2z-dsa-course/strivers-a2z-dsa-course-sheet-2/_blank)
13. [Merge Overlapping Subintervals](https://takeuforward.org/data-structure/merge-overlapping-sub-intervals/" \t "https://takeuforward.org/strivers-a2z-dsa-course/strivers-a2z-dsa-course-sheet-2/_blank)
14. [Merge two sorted arrays without extra space](https://takeuforward.org/data-structure/merge-two-sorted-arrays-without-extra-space/" \t "https://takeuforward.org/strivers-a2z-dsa-course/strivers-a2z-dsa-course-sheet-2/_blank)
15. [Find the repeating and missing number](https://takeuforward.org/data-structure/find-the-repeating-and-missing-numbers/" \t "https://takeuforward.org/strivers-a2z-dsa-course/strivers-a2z-dsa-course-sheet-2/_blank)
16. [Reverse Pairs](https://takeuforward.org/data-structure/count-reverse-pairs/" \t "https://takeuforward.org/strivers-a2z-dsa-course/strivers-a2z-dsa-course-sheet-2/_blank)
17. [Maximum Product Subarray](https://takeuforward.org/data-structure/maximum-product-subarray-in-an-array/" \t "https://takeuforward.org/strivers-a2z-dsa-course/strivers-a2z-dsa-course-sheet-2/_blank)

BINARY SEARCH

1. Search in Rotated Sorted Array II Tc = O(log n)
2. Koko Eating Bananas
3. **Median in a row-wise sorted Matrix**
4. [Find kth element of two sorted arrays](https://takeuforward.org/data-structure/k-th-element-of-two-sorted-arrays/" \t "https://takeuforward.org/strivers-a2z-dsa-course/strivers-a2z-dsa-course-sheet-2/_blank)
5. [Aggressive Cows](https://takeuforward.org/data-structure/aggressive-cows-detailed-solution/" \t "https://takeuforward.org/strivers-a2z-dsa-course/strivers-a2z-dsa-course-sheet-2/_blank)

GRAPH

1. [Cycle Detection in unirected Graph (bfs)](https://takeuforward.org/data-structure/detect-cycle-in-an-undirected-graph-using-bfs/" \t "https://takeuforward.org/strivers-a2z-dsa-course/strivers-a2z-dsa-course-sheet-2/_blank)
2. [Number of Enclaves](https://takeuforward.org/graph/number-of-enclaves/" \t "https://takeuforward.org/strivers-a2z-dsa-course/strivers-a2z-dsa-course-sheet-2/_blank)
3. Number of Distinct Islands
4. Find eventual safe states using dfs
5. Cycle Detection in Directed Graph (DFS) kahn algo
6. Alien dictionary
7. Word ladder – 1(imp)
8. Word ladder – 2(imp)
9. Print Shortest Path - Dijkstra's Algorithm
10. Cheapest flights within k stops
11. Number of ways to arrive at destination
12. Prim’s Algorithm
13. Making a Large Island
14. Most stones removed with same rows or columns

**LinkedList**

1. [Find the starting point in LL](https://takeuforward.org/data-structure/starting-point-of-loop-in-a-linked-list/" \t "https://takeuforward.org/strivers-a2z-dsa-course/strivers-a2z-dsa-course-sheet-2/_blank) in O(n) time
2. Segrregate odd and even nodes in LL in O(n) time
3. [Reverse LL in group of given size K](https://takeuforward.org/data-structure/reverse-linked-list-in-groups-of-size-k/" \t "https://takeuforward.org/strivers-a2z-dsa-course/strivers-a2z-dsa-course-sheet-2/_blank)
4. Sort LL
5. Segrregate odd and even nodes in LL
6. [Check if LL is palindrome or not](https://takeuforward.org/data-structure/check-if-given-linked-list-is-plaindrome/" \t "https://takeuforward.org/strivers-a2z-dsa-course/strivers-a2z-dsa-course-sheet-2/_blank)
7. [Rotate a LL](https://takeuforward.org/data-structure/rotate-a-linked-list/" \t "https://takeuforward.org/strivers-a2z-dsa-course/strivers-a2z-dsa-course-sheet-2/_blank)
8. [Clone a Linked List with random and next pointer](https://takeuforward.org/data-structure/clone-linked-list-with-random-and-next-pointer/" \t "https://takeuforward.org/strivers-a2z-dsa-course/strivers-a2z-dsa-course-sheet-2/_blank)
9. [Flattening of LL](https://takeuforward.org/data-structure/flattening-a-linked-list/" \t "https://takeuforward.org/strivers-a2z-dsa-course/strivers-a2z-dsa-course-sheet-2/_blank)

**String**

1. Longest Palindromic Substring O(n^2) and Manacher's Algorithm O(n)

**Recursion**

1. Pow(x, n) in log(n) time
2. Sort a stack using recursion
3. Combination Sum II
4. [Subset Sum-II](https://takeuforward.org/data-structure/subset-ii-print-all-the-unique-subsets/" \t "https://takeuforward.org/strivers-a2z-dsa-course/strivers-a2z-dsa-course-sheet-2/_blank)
5. Permutation in String(leetcode)
6. N-Queens
7. Sudoko Solver
8. Permutation Sequence

**DP**

1. **Minimum/Maximum Falling Path Sum**
2. **3-d DP : Ninja and his friends**
3. Partition Set Into 2 Subsets With Min Absolute Sum Diff (DP- 16)
4. space optimization using only 1 row (i.e. only using prev) 0/1 knapsack