***Data Structures***

***Stack:***

1. Array Implementation of Stack.
2. Linked List implementation of Stack.
3. Array Implementation of stack without using top pointer.
4. Design a stack that returns the minimum element in constant time.
5. Implement two stacks in a single array.
6. Reverse a stack using recursion.
7. Reverse a string using stack.
8. Implement a max stack.

***Linked List:***

1. Singly Linked List Implementation.
2. Doubly Linked List Implementation.
3. Circular Linked List Implementation.
4. Reverse of a Singly Linked List.
5. *Reverse of a Doubly Linked List.*
6. *Polynomials Operations.*
7. *WAP to Inserting a node on a linked list in a sorted order.*
8. *WAP to merge two sorted linked list.*
9. *Flattering a linked list.*
10. *Remove Duplicate elements from linked list.*
11. *Rotate a linked list.*
12. *Remove all occurrences of a specific element from linked list.*
13. *Find the middle node of a linked list.*

***Sliding Window Technique:***

1. *WAP to find maximum sum subarray of size K.*
2. *WAP to find first negative number in every window of size K.*
3. *WAP to count occurrences of anagrams.*
4. *WAP to find largest subarray of sum K.*