1)Explain the Binary Search algorithm. Write its pseudocode and discuss its time complexity.

2)Describe the Quick Sort algorithm. Write its pseudocode and analyze its time complexity.

3)Explain why a linear queue can lead to wasted space and how a circular queue helps overcome this by reusing space.

4)Write a function in C to insert a node at the beginning of a singly linked list.discuss the time complexity of this operation..

5)Explain the difference between sequential and binary search. Discuss the advantages and disadvantages of each

6)What is stack in data structure? Write an algorithm for infix expression to postfix expression evaluation.

7) Provide an algorithm to evaluate a postfix expression using a stack and explain how each step processes the operators and operands.

8)Explain the concept of recursion and how a stack is used to manage recursive function calls in memory.

9) Outline the steps involved in converting an infix expression to postfix using a stack for operators.

10) What is queue in data structure?Write pseudo c code for enqueue and dequeue operation on queue.

11)What is the priority queue?Explain queue as an ADT.

12) Define a queue data structure. Discuss the differences between a circular queue and a regular queue, and provide a use case for each

13)Write an algorithm for postfix expression evaluation using a stack. Explain each step with an example expression

14)Outline the steps involved in converting an infix expression (e.g., A + B * C) to postfix using a stack for operators

15)Define a stack as an ADT in data structure.

16) Define a queue data structure. Discuss the differences between a circular queue and a regular queue, and provide a use case for each

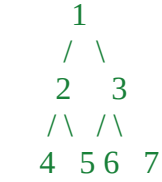
17)Write an algorithm for prefix expression evaluation using a stack. Explain each step with an example expression

18)Describe a priority queue and its applications. Compare its implementation using an array versus a binary heap

19)Write a recursive algorithm to find the minimum and maximum values in a BST..

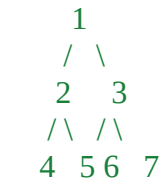
20)Explain Preorder traversal on a binary tree.

Here's an example of a binary tree:

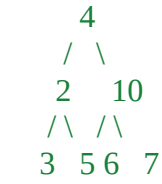


Given a sample binary tree, demonstrate the traversal steps..

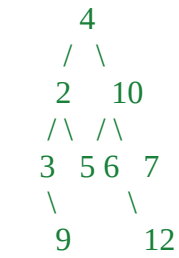
21) Explain step by step Preorder traversal on a binary tree.



22) Explain step by step Postorder traversal given a binary tree.



24) Balanced given AVL Tree with different rotations.



25) Explain how null pointers are used in a threaded binary tree. Discuss their advantages in terms of memory usage and traversal efficiency

26)Using a min-heap, sort the array [42, 29, 18, 14, 35, 8, 22]. Show the array transformation at each step

29)What is BST? Explain Recursive algorithm for binary tree.
30)Explain how efficient use of null pointer in threaded binary tree?explain with a neat diagram

31)What is the difference between an array and a linked list?

32)Explain the working of a stack data structure. Where is it used in real-world applications?

33)How does a queue differ from a deque (double-ended queue)?

34)What is a binary tree? How is it different from a binary search tree?

35)Explain the concept of hashing. What are hash collisions, and how can they be resolved?

36)Write an algorithm to reverse a singly linked list.

37)How can you find the middle element of a linked list in one traversal?

38)What is the time complexity of searching in a binary search tree (BST)?

39)Describe the difference between Depth First Search (DFS) and Breadth First Search (BFS) in graphs.

40)What is the difference between Merge Sort and Quick Sort in terms of time complexity and working?

41)What is a heap data structure? How is it implemented?

42)Explain the concept of dynamic programming with an example.

43)How would you detect a cycle in a directed graph?

44)Write an algorithm to find the shortest path in a graph using Dijkstra's algorithm.

45)What is the difference between a greedy algorithm and a dynamic programming approach?

46)How would you find the k-th largest element in an unsorted array?

47)Given a string, write an algorithm to check if it is a palindrome using a stack.

48)How would you implement a min-stack that supports O(1) retrieval of the minimum element?

49)Write an algorithm to find the maximum sum subarray using Kadane’s Algorithm.

50)Design an algorithm to check whether a binary tree is a valid binary search tree (BST).