Omgamganapathayenamaha

Omgamganapathayenamaha

Omgamganapathayenamaha

Omgamganapathayenamaha

Chapter 1: Introduction

1. Big-O Notation
2. Omega Notation
3. Theta Notation
4. Asymptotic Curve
5. Asymptotic Analysis
6. Best Case, Average Case and Worst Case
7. Master Theorem for Recurrences

Chapter 2: Recursion and Backtracking

1. Towers of Hanoi
2. Finding whether a given array is sorted
3. Permutations of a string
4. Combinations of a string
5. Generating Binary Strings
6. Generating k-ary Strings
7. Find the largest region of connected 1’s in a given matrix of 0’s and 1’s
8. Given a matrix of 0’s and 1’s, find a path from position (0,0) to (m-1, n-1)
9. Hamiltonian Cycles
10. N-Queens
11. Graph Coloring
12. The Knapsack problem
13. Generalized Strings

Chapter 3: Linked Lists

1. Singly Linked List Implementation
2. Singly Circular Linked List Implementation
3. Doubly Linked List Implementation
4. Doubly Circular Linked List Implementation
5. Unrolled Linked Lists: Nodes are grouped to form blocks and there will be sqrt(n) blocks. We can find kth element in O(sqrt(n)) complexity
6. Skip Lists: Sorted Linked Lists with multiple levels. Instead of traversing sequentially, we can skip few nodes in between to gain O(logn) performance
7. Find the Middle Element in one scan
8. Find the kth element from the end in one scan
9. Reverse the list
10. Find whether loop exists in the list : Floyd cycle finding algorithm
    1. If loop exists, find the start node of the loop
    2. If loop exists, find the loop length
11. Given 2 lists of size m and n which are merging into one linked list, find the merging point
12. Reverse blocks of k nodes in a list
13. Find Modular Node
14. Find sqrt(n)th node
15. Find Fractional Node
16. Addition of Lists: List1 contains say 1-2-3 and list2 contains say 9->8. The result should be a list containing 2-2-1

Chapter 4: Stacks

1. Stack Implementation
2. Balancing of Symbols
3. Infix to Postfix Conversion
4. Evaluation of Postfix expression
5. Converting Postfix expression to Expression Trees (used by compilers)
6. Design a Stack such that get\_minimum() is O(1)
7. Given 2 lists of size m and n which are merging into one linked list, find the merging point
8. Finding Spans
9. Largest Rectangle under Histogram – Judges’ Algorithm
10. Given an array, find NextNearestGreater element to the right of it
11. Given a list say A1,A2,A3,A4,A5,A6, generate a list by changing the order as A1,A6,A2,A5,A3,A4

Chapter 5: Queues

1. Queue Implementation
2. Reverse the queue (use Stack)
3. Implement a Stack using 2 Queues
4. Implement a Queue using 2 Stacks
5. Implement a Doubly-Ended Queue
6. Given a queue of integers, create an interleaving Queue
7. Give an integer k and a Queue of integers, reverse the first k integers in the Queue keeping other elements untouched
8. Producer Consumer problem using Threads and a Queue
9. Given a String, check whether it is a palindrome (try using deque)