MERCY TECHNOLOGIES 79758 55775

DATA STRUCTURES USING JAVA

1. Introduction to Data Structures

2. Memory Segments Introduction with simple programs.
   a. Heap Memory (Dynamic Memory)
   b. Stack Memory (Local Memory)
   c. Global Memory (Code Area)

3. Finding Time Complexity and Space Complexity using sample code.

4. Iterative vs Recursive approach for Factorial and Fibonacci programs.

5. Comparing Time Complexity and Space Complexity in Iterative and Recursive approaches.

6. Some Logarithmic formulae.

7. **Linked List**
   a. Linked List Object
   b. Traversal
   c. Insert
   d. Insert First
   e. Insert Last
   f. Insert at particular point (Index)
   g. Delete
   h. Delete First
   i. Delete Last
   j. Delete at particular point (Index)
   k. Count number of Nodes
   l. Find middle element
   m. Find if linked list is circular linked list or not
   n. Reverse a LinkedList

8. **Stacks and Queues**
   a. push()
   b. pop()
   c. enque()
   d. deque()
   e. Create a stack using LinkedList
   f. Create a queue using LinkedList
   g. Create a stack using Array
   h. Create a queue using Array
   i. Few more interesting questions to solve
   j. Space and Time complexity of all the above-mentioned topics

9. **Trees**
   a. Binary Trees
      i. BinaryTree Object
      ii. Internal and External Nodes
      iii. Degree of a Tree
      iv. Height of a Tree

   b. Types of Binary Trees
      i. Strict Binary Tree
      ii. Full Binary Tree
      iii. Complete Binary Tree

   c. Types of Binary Tree Traversals
      i. Pre-Order Traversal
      ii. In-Order Traversal
      iii. Post-Order Traversal
      iv. Level-Order Traversal

   d. Height of a Binary Tree
   e. Number of elements in a Binary Tree
   f. Max element in a Binary Tree
   g. Number of Leaves in a Binary Tree

10. **Graphs**
   a. Permutations vs Combinations
   b. Edge
   c. Vertex
   d. Trees vs Graphs
   e. In degree
   f. Out degree
   g. What is Adjacency Matrix & Adjacency List?
   h. BFS
   i. DFS
   j. Real time problems with solution.