Test Preparations

- Tree-like data structures
    - Advantages of tree structure
    - Linear data structure
    - Real-world application where tree structures are preferred.
- Binary search trees
    - Characteristics and applications
    - Efficient search and insertion operations
- Explain the concept of self-balancing tree and their significance.
    - Compare AVL trees, scapegoat trees, and red-black trees in terms of balancing criteria, insertion and deletion complexity, and their typical applications.
- Explain graph data structures and their real-world application.
    - Explain different types of graphs and their representations.
- Explain the concept of hash tables and their role in efficient data storage and retrieval.
    - Discuss collision resolution techniques used in hash tables and their impact on performance.
- Implement a binary search tree – insertion, search, and perform in-order traversal.
- Implement heap data structure – insertion and heapify operations.
- Implement AVL tree – insertion and deletion.
- Implement scapegoat tree – insert methods.
- Implement red-black tree - insert methods.
- Implement graph data structure – add vertices and edges.
- Implement hash tables using separate chaining collision resolution techniques – insert, delete, and search for key-value pairs.
- Implement hash table data structure using linear probing collision resolution technique – insert, delete, and search key-value pairs.
- Theoretical computer language in coding?