0.1 1333203 DSA Winter 2023

0.1.1 Q1a: Define linked list. List different types of linked list. (03 marks)

Ans 1a: A linked list is a dynamic data structure consisting of a sequence of elements, where each element (called a node) contains data and a reference (or link) to the next element in the sequence. Unlike arrays, linked lists do not store elements in contiguous memory locations, allowing for efficient insertion and deletion operations.

Key characteristics of linked lists: - Dynamic size: Can grow or shrink during program execution - Non-contiguous memory allocation: Elements can be stored anywhere in memory - Efficient insertion and deletion: O(1) time complexity for operations at the beginning or end

Different types of linked lists:

1. **Singly Linked List**:
    - Each node contains data and a single reference to the next node
    - Last node points to NULL, indicating the end of the list
2. **Doubly Linked List**:
    - Each node contains data and two references: one to the next node and one to the previous node
    - Allows traversal in both directions
3. **Circular Linked List**:
    - Similar to singly linked list, but the last node points back to the first node
    - Forms a closed loop
4. **Circular Doubly Linked List**:
    - Combines features of doubly linked and circular linked lists
    - Last node's next pointer points to the first node, and first node's previous pointer points to the last node
5. **Header Linked List**:
    - Contains a special header node at the beginning
    - Header node may store metadata about the list (e.g., size, pointers to first and last elements)

0.1.2 □□□□□□□1□: □□□□□□□ □□□□□□□□□ □□□□□□□□□□□ □□□□. □□□□□□□ □□□□□□□□□□□ □□□□□□□□□□ □□□□□□ □□ □□□□□ □□□□. (□□ □□□□)

□□□□□1□: linked list□ □□ □□□□□□□□□□□ □□□□□ □□□□□□□□□□□□ □□ □□□□□□ □□□□□□□□□□□□□□□ □□□□□ □□□□ □□, □□□□□□ □□□□□ □□□□□□□□□□ (□□□□□ node □□□□□□□ □□) □□□□□ □□□□ □□□□□□□□□ □□□□□□ □□□□□□□□□□□□□□ □□□□□□□ (□□□□□ link) □□□□□□ □□. □□□□□□□ □□□□□□□, linked lists [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] [illegible], [illegible] insertion [illegible] deletion [illegible] [illegible] [illegible] [illegible].

linked lists [illegible] [illegible] [illegible]: - [illegible] [illegible]: [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] - [illegible]-[illegible] [illegible] [illegible]: [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] - [illegible] insertion [illegible] deletion: [illegible] [illegible] [illegible] [illegible] [illegible] O(1) [illegible] [illegible]

[illegible] [illegible] linked lists:

1. **Singly Linked List**:
   - [illegible] node [illegible] [illegible] [illegible] node [illegible] [illegible] [illegible] [illegible] [illegible] [illegible]
   - [illegible] node NULL [illegible] [illegible] [illegible] [illegible], [illegible] [illegible] [illegible] [illegible] [illegible]
2. **Doubly Linked List**:
   - [illegible] node [illegible] [illegible] [illegible] [illegible] [illegible] [illegible]: [illegible] [illegible] node [illegible] [illegible] [illegible] [illegible] node [illegible]
   - [illegible] [illegible] [illegible] [illegible] [illegible] [illegible]
3. **Circular Linked List**:
   - Singly Linked List [illegible] [illegible], [illegible] [illegible] node [illegible] node [illegible] [illegible] [illegible] [illegible] [illegible]
   - [illegible] [illegible] [illegible] [illegible]
4. **Circular Doubly Linked List**:
   - Doubly Linked [illegible] Circular Linked Lists [illegible] [illegible] [illegible] [illegible]
   - [illegible] node [illegible] next pointer [illegible] node [illegible] [illegible] [illegible] [illegible], [illegible] [illegible] node [illegible] previous pointer [illegible] node [illegible] [illegible] [illegible] [illegible]
5. **Header Linked List**:
   - [illegible] [illegible] [illegible] header node [illegible] [illegible]
   - Header node [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] ([illegible]., [illegible], [illegible] [illegible] [illegible] [illegible] pointers)