Q5. Understand Linked Lists:

There are two main types of linked lists:

a) Singly Linked List: Each node contains data and a reference (link) to the next node in the sequence. This is what we've implemented in our example.

b) Doubly Linked List: Each node contains data and two references (links) - one to the next node and one to the previous node.

Analysis:

Time complexity of operations:

Add: O(n) - we need to traverse to the end of the list to add a new task

Search: O(n) - we may need to traverse the entire list to find a task

Traverse: O(n) - we need to visit each node once

Delete: O(n) - we may need to traverse the entire list to find the task to delete

Advantages of linked lists over arrays for dynamic data:

Dynamic size: Linked lists can grow or shrink in size during runtime without needing to reallocate memory.

Efficient insertions and deletions: Adding or removing elements (especially at the beginning) is more efficient in linked lists (O(1)) compared to arrays (O(n)).

No wasted space: Linked lists use exactly as much memory as they need.

Flexibility: Linked lists can easily grow or shrink in size, making them ideal for applications where the number of items is unknown or changes frequently.