**Exercise 5: Task Management System**

1. Explain the different types of linked lists (Singly Linked List, Doubly Linked List).
   * Singly Linked List
     1. Structure: A node contains data and a pointer, which refers the next node in the list.
     2. Traversal: It only allows to traverse in the forward direction.
     3. Memory: In comparison to the memory allocated in the case of a doubly linked list, the singly linked list has lesser memory overhead as here, each node has got only one reference.
   * Doubly Linked List
     1. Structure: In doubly linked lists, for every node, there is a pointer associated with the next node and a pointer concerning the previous node.
     2. Traversal: The doubly linked list allows reversing the process of both forwards and backward traversal.
     3. Memory: The memory overhead is higher than that in a singly linked list by definition as each node has 2 references.
2. Analyze the time complexity of each operation.
   * Add Task: O(n) – The addition of a task can only be done by traversing through the whole list up to its end. This is an operation whose time complexity is linear and depends on the number of tasks being added.
   * Search Task: O(n) – Searching for a task means going through a list of tasks, which will take linear time for the number of tasks.
   * Traverse Tasks: O(n) –Traversing all tasks requires going through the entire the list of element, in linear time.
   * Delete Task: O(n) — Deletion requires linear time searching for the task, in question, to delete and constant time updating the pointers.
3. Discuss the advantages of linked lists over arrays for dynamic data.
   * Dynamic Size: The size of linked lists can be changed dynamically; hence, they can be used for applications where the number of elements needs to be changed frequently.
   * Efficient Insertions/Deletions: Insertions and deletions are more efficient than arrays in a linked list, especially when one is inserting/removing elements in the middle of the list.
   * Memory Utilization: In link list require not continuous block of memory so to avoid problems of memory fragmentation.