Task Management System

**Understand Linked Lists**

**Types of Linked Lists:**

1. **Singly Linked List**:
   * **Structure**: Consists of nodes where each node contains data and a reference to the next node.
   * **Traversal**: Can only traverse in one direction, from the head to the end of the list.
   * **Insertion/Deletion**: Easier to insert or delete nodes at the beginning of the list but requires traversal to access other positions.
2. **Doubly Linked List**:
   * **Structure**: Each node contains data, a reference to the next node, and a reference to the previous node.
   * **Traversal**: Can traverse in both directions (forward and backward).
   * **Insertion/Deletion**: Easier to insert or delete nodes at both ends and in between, as it has references to both previous and next nodes.

**Time Complexity:**

* **Add**: O(n), as you may need to traverse to the end of the list.
* **Search**: O(n), requires traversal of the list to find the task.
* **Traverse**: O(n), requires visiting each node in the list.
* **Delete**: O(n), as you need to find the node first, and then remove it, adjusting pointers as necessary.

**Advantages of Linked Lists Over Arrays for Dynamic Data:**

* **Dynamic Size**: Linked lists can easily grow and shrink in size, as they don't require a fixed size like arrays.
* **Efficient Insertions/Deletions**: Inserting or deleting elements in the middle of a linked list is more efficient compared to arrays, as there’s no need to shift elements.
* **Memory Utilization**: Linked lists use memory efficiently when the size of the data set changes frequently, as they allocate memory only for the required nodes.