4 Lecture - CS301

Important Subjective

1. **What is a Linked List?**
   Answer: A Linked List is a linear data structure that consists of a sequence of nodes, where each node contains data and a pointer to the next (and possibly the previous) node in the list.

2. **What is the difference between a singly linked list and a doubly linked list?**
   Answer: In a singly linked list, each node has a pointer to the next node in the list, while in a doubly linked list, each node has a pointer to both the next and previous nodes in the list.

3. **What is a head pointer and a tail pointer in a Linked List?**
   Answer: The head pointer points to the first node in the list, while the tail pointer points to the last node in the list.

4. **How is a new node inserted at the beginning of a singly linked list?**
   Answer: To insert a new node at the beginning of a singly linked list, a new node is created and its next pointer is set to the current head of the list. The head pointer is then updated to point to the new node.

5. **How is a new node inserted at the end of a singly linked list?**
   Answer: To insert a new node at the end of a singly linked list, a new node is created and its next pointer is set to NULL. The next pointer of the current last node is updated to point to the new node, and the tail pointer is updated to point to the new node.

6. **How is a node deleted from a singly linked list?**
   Answer: To delete a node from a singly linked list, the next pointer of the previous node is updated to point to the next node in the list. The memory occupied by the deleted node is then freed.

7. **What is a sentinel node in a Linked List?**
   Answer: A sentinel node is a dummy node that is added to the beginning or end of a Linked List to simplify certain operations, such as inserting or deleting nodes at the beginning or end of the list.

8. **What is the time complexity of searching for an element in a Linked List?**
   Answer: The time complexity of searching for an element in a Linked List is O(n), where n is the number of nodes in the list.

9. **How is a Linked List traversed recursively?**
   Answer: A Linked List can be traversed recursively by starting at the head of the list and calling a function that takes the current node as an argument and recursively calls itself with the next node in the list.

10. **What is a circular Linked List?**
    Answer: A circular Linked List is a Linked List where the last node points back to the first node, creating a circular structure. This can be either a singly linked or doubly linked list.