Summary

Sessions (08-02-2023)

- **Get Operation :**

```
int LinkedList::Get(int index) {
    Node *last = head;

    for (int i = 0; i <= index; i++) {
        last = last -> next;
    }
    return last -> data;
}
```

Time complexity in Best case scenario : O(1)

Time complexity in Worst case scenario : O(n)

- **Update Operation :**

```
void LinkedList::Update(int newData, int index) {
    Node *last = head;

    for (int i = 0; i <= index; i++) {
        last = last -> next;
    }

    last -> data = newData;
    cout << "Linked List Is Updated...." << endl;
}
```

Time complexity in Best case scenario : O(1)

Time complexity in Worst case scenario : O(n)

- **isEmpty() operation :**

```
// isEmpty operation
void LinkedList::isEmpty() {
    if(head -> next == NULL)
        cout << "Linked List Is Empty..." << endl;
    else
        cout << "Linked List Is Not Empty..." << endl;
}
```

- **Delete Operation :**

```
void LinkedList::Delete(int index) {
    Node *last = head;

    for (int i = 0; i < index; i++) {
        last = last -> next;
    }

    Node *temp = last -> next;
    last -> next = temp -> next;

    delete temp;
}
```