Minor : Perform Deletion at an Index on an LinkedList

-Prakhar Gupta

```cpp
#include <iostream>

using namespace std;


struct node {

    int data;

    node* next;

};
void insert(node **start, int data) {

    node* nnode = new node;

    nnode->data = data;

    nnode->next = *start;

    *start = nnode;

}
node* deleteN(node* start, int index) {

    if (start == NULL) {

        cout << "LinkedList is empty\n";

        return start;

    }

    if (index == 1) {

        node* temp = start;

        start = start->next;

        delete temp;

        return start;

    }

    node* current = start;

    node* previous = NULL;

    for (int i = 1; i < index && current != NULL; i++) {

        previous = current;

        current = current->next;

    }

    if (current == NULL) {
```

```cpp
            cout << "invalid index\n";

            return start;

        }

        previous->next = current->next;

        delete current;

        return start;

    }

    void print(node* start) {

        node* temp = start;

        while (temp != NULL) {

            cout <<" "<<temp->data;

            temp = temp->next;

        }

        cout << "\n";

    }

    int main() {

        node* start = NULL;

        int n, data, index;

        cout << "Enter the number of nodes: ";

        cin >> n;

        cout << "Enter the nodes: ";

        for (int i = 0; i < n; i++) {

            cin >> data;

            insert(&start, data);

        }

        cout << "Enter the index to delete a node: ";

        cin >> index;

        start = deleteN(start, index);

        cout << "After deleting of node at index " << index << " ll is: ";

        print(start);

        return 0;
```

```
Enter the number of nodes: 5
Enter the nodes: 1
2
3
4
5
Enter the index to delete a node: 3
After deleting of node at index 3 ll is:  5 4 2 1
}
```