1 . Write a function to get the nth node from the end of the linked list.

AZHAR ALI

CMS : 023-23-0314

**Sec A**

ASSIGment 1

DSA

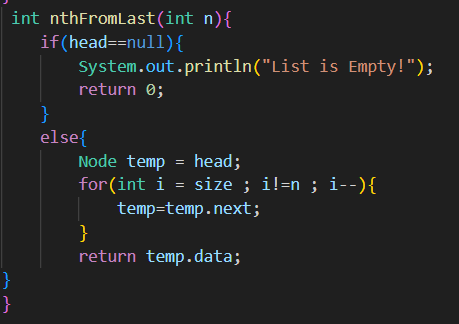
Function name: int nthFromLast(int n);

Case-1: (List Empty) Head=Null then return LIST\_EMPTY

Case-2: (List Non-Empty) Head != Null then return nth element from the end of list

Example

Input: 10 -> 20 -> 30 -> 40 -> 50, n = 2

Output: 40 (From the last, second node conatins the data 40)

Write a function to sort the given single linked list. (Don’t swap the data present in the nodes, swap the nodes itself.)

Function name: void sort();

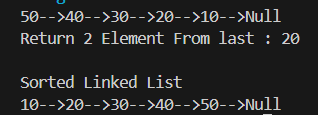
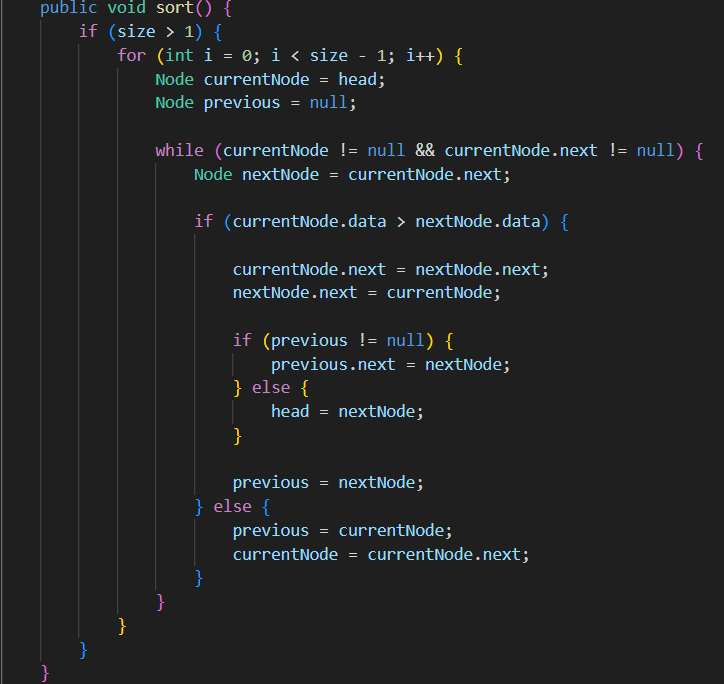
Case-1: (List Empty) Head=Null then return LIST\_EMPTY

Case-2: (List Non-Empty) Head != Null then swap the nodes to sort them

Example

Input: 50 -> 40 -> 30 -> 20 -> 10

Output: 10 -> 20 -> 30 -> 40 -> 50



1. Write a function to reverse the single linked list.

Function name: void reverse();

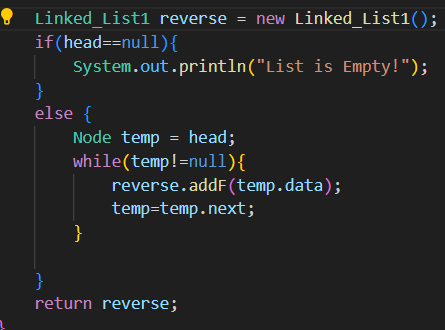
Case-1: (List Empty) Head=Null then return LIST\_EMPTY

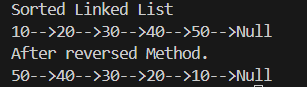
Case-2: (List Non-Empty) Head != Null then reverse the list

Example

Input: 50 -> 40 -> 30 -> 20 -> 10

Output: 10 -> 20 -> 30 -> 40 -> 50





1. Write a function to remove the duplicates data present in the single linked list.

Function name: void removeDuplicates();

Case-1: (List Empty) Head=Null then return LIST\_EMPTY

Case-2: (List Non-Empty) Head != Null then remove duplicate elements

Example

Input: 5 -> 3 -> 4 -> 5 -> 2 -> 1 -> 4 -> 5 -> 3

Output: 5 -> 3 -> 4 -> 2 -> 1

