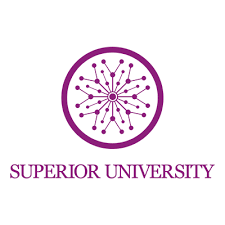
**TASK NO 5**

****

**Name:  
M. Zuhaib Anwar**

**Class :**

**SE – 3A**

**Roll No:**

**SU92-BSSEM-S24-005**

#include <iostream>

class Node {

public:

int data;

Node\* next;

Node(int value) : data(value), next(nullptr) {}

};

class SinglyLinkedList {

private:

Node\* head;

public:

SinglyLinkedList() : head(nullptr) {}

void insert(int value) {

Node\* newNode = new Node(value);

if (!head) {

head = newNode;

} else {

Node\* temp = head;

while (temp->next) {

temp = temp->next;

}

temp->next = newNode;

}

}

void displayFirstNode() {

if (head) {

std::cout << "First Node: " << head->data << std::endl;

}

}

void displayLastNode() {

if (head) {

Node\* temp = head;

while (temp->next) {

temp = temp->next;

}

std::cout << "Last Node: " << temp->data << std::endl;

}

}

void displayNthNode(int n) {

Node\* temp = head;

int count = 0;

while (temp) {

if (count == n) {

std::cout << "Nth Node: " << temp->data << std::endl;

return;

}

count++;

temp = temp->next;

}

std::cout << "Node not found." << std::endl;

}

void displayCentreNode() {

Node\* slow = head;

Node\* fast = head;

if (head) {

while (fast && fast->next) {

slow = slow->next;

fast = fast->next->next;

}

std::cout << "Centre Node: " << slow->data << std::endl;

}

}

};

int main() {

SinglyLinkedList list;

list.insert(1);

list.insert(2);

list.insert(3);

list.insert(4);

list.insert(5);

list.displayFirstNode();

list.displayLastNode();

list.displayNthNode(2);

list.displayCentreNode();

return 0;

}

OUTPUT:

