**NAME : MALIHA SHAHID**

**ROLL NO:SU92-BSSEM-S24-071**

**LAB 8**

**Question #01**

#include <iostream>

using namespace std;

class ListNode {

public:

int value;

ListNode\* nextNode;

ListNode(int v) {

value = v;

nextNode = NULL;

}

};

class SingleLinkedList {

public:

ListNode\* firstNode;

SingleLinkedList() {

firstNode = NULL;

}

void append(int val) {

ListNode\* newNode = new ListNode(val);

if (firstNode == NULL) {

firstNode = newNode;

}

else {

ListNode\* temp = firstNode;

while (temp->nextNode != NULL) {

temp = temp->nextNode;

}

temp->nextNode = newNode;

}

}

void showList() {

ListNode\* temp = firstNode;

while (temp != NULL) {

cout << temp->value << " -> ";

temp = temp->nextNode;

}

cout << "NULL" << endl;

}

};

ListNode\* mergeLists(ListNode\* list1, ListNode\* list2) {

if (!list1) return list2;

if (!list2) return list1;

if (list1->value < list2->value) {

list1->nextNode = mergeLists(list1->nextNode, list2);

return list1;

}

else {

list2->nextNode = mergeLists(list1, list2->nextNode);

return list2;

}

}

int main() {

SingleLinkedList sll1, sll2;

sll1.append(6);

sll1.append(5);

sll1.append(4);

sll2.append(3);

sll2.append(2);

sll2.append(1);

cout << "First Linked List: ";

sll1.showList();

cout << "Second Linked List: ";

sll2.showList();

SingleLinkedList mergedSLL;

mergedSLL.firstNode = mergeLists(sll1.firstNode, sll2.firstNode);

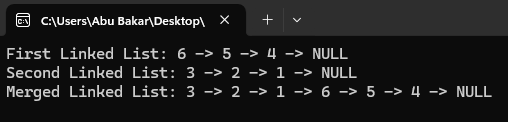
cout << "Merged Linked List: ";

mergedSLL.showList();

return 0;

}

**OUTPUT**

****

**EXPLANATION**

* The ListNode class defines a node with a value and a pointer to the next node.
* The SingleLinkedList class manages a linked list with functions to add and display nodes.
* The append(int val) function adds a new node at the end of the list.
* The showList() function prints the linked list.
* The mergeLists() function recursively merges two sorted linked lists.
* In main(), two linked lists (sll1 and sll2) are created and populated with values.
* The two lists are displayed, then merged using mergeLists().