**Assignment no #1**

**Subject :Data Structures**

**Name :Muhammad Shaheer.**

**Roll no: 20p-0480.**

**Question** : Write a function named swapNodes that swaps two nodes of a singly linked list by updating

the next pointers of the nodes. The function should take two integer parameters specifying

the position of the nodes to be swapped. The prototype of the function is as follows: [CLO 1]

vo id swapNodes ( i n t lPo s , i n t rPo s ) ;

where lPos, rPos are positions of the left and right nodes respectively. The following function

call will swap nodes at position 2 and 5.

swapNodes (2 , 5 ) ;

**Code**:

#include <iostream>

using namespace std;

struct Node {

    int data;

    Node\* next;

};

void swapNodes(int lPos, int rPos, Node\* head) {

    if (lPos == rPos) {

        return;

    }

    Node\* prevL = NULL;

    Node\* currL = head;

    for (int i = 1; i < lPos && currL != NULL; i++) {

        prevL = currL;

        currL = currL->next;

    }

    Node\* prevR = NULL;

    Node\* currR = head;

    for (int i = 1; i < rPos && currR != NULL; i++) {

        prevR = currR;

        currR = currR->next;

    }

    if (currL == NULL || currR == NULL) {

        return;

    }

    if (prevL != NULL) {

        prevL->next = currR;

    } else {

        head = currR;

    }

    if (prevR != NULL) {

        prevR->next = currL;

    } else {

        head = currL;

    }

    Node\* temp = currR->next;

    currR->next = currL->next;

    currL->next = temp;

}

int main() {

    // create a sample linked list

    Node\* head = new Node{1, new Node{2, new Node{3, new Node{4, new Node{5, NULL}}}}};

    // print the original linked list

    Node\* curr = head;

    cout<<"Before  Linked List"<<endl;

    while (curr != NULL) {

        cout <<curr->data << "  ";

        curr = curr->next;

    }

    cout << endl;

cout << endl;

    // swap nodes at positions 2 and 5

    swapNodes(2, 5, head);

    // print the modified linked list

    cout<<"After  the  swaping link list "<<endl;

    curr = head;

    while (curr != NULL) {

        cout << curr->data << " ";

        curr = curr->next;

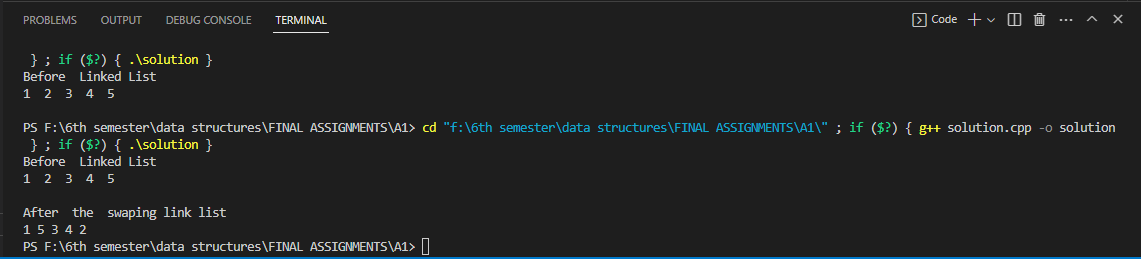
    }

    cout << endl;

    return 0;

}

**Output**:



THE END\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_