1. Write a function to reverse a singly linked list.The function should take the head of the list and return the new head of the reversed list.

// creating the linked list

#include <iostream>

using namespace std;

class ListNode{

public:

int data;

ListNode\* next;

ListNode\* head;

ListNode(int data){

this->data=data;

this->next=NULL;

}

};

// function to create the linked list

void addIntoList(ListNode\* &head, int data){

ListNode\* temp= new ListNode(data);

if(head==NULL){

head=temp;

return ;

}

ListNode\* curr=head;

while(curr->next){

curr=curr->next;

}

curr->next=temp;

return;

}

// function to print the linked list

void printList(ListNode\* head){

ListNode\* temp=head;

if(temp==NULL)

cout<<"list is empty";

while(temp){

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

temp=temp->next;

}

cout<<endl;

}

// function to reverse the linked list

ListNode\* reverseList(ListNode\* &head){

if(head==NULL || head->next == NULL){

return head;

}

ListNode\* newHead= reverseList(head->next);

head->next->next=head;

head->next=NULL;

return newHead;

}

// main function

int main(){

ListNode\* head=NULL;

addIntoList(head,9);

addIntoList(head,5);

addIntoList(head,6);

addIntoList(head,9);

addIntoList(head,11);

addIntoList(head,3);

//printing list before reverse;

printList(head);

ListNode\* newHead = reverseList(head);

head=newHead;

cout<<"new head of reversed linked list: "<<head->data<<endl;

//printing list after reverse;

cout<<"reversed list is"<<endl;

printList(head);

}

OUTPUT

