Rajalakshmi Engineering College

Name: subhashri bala
Email: 240801337@rajalakshmi.edu.in
Roll no: 2116240801337
Phone: 7418182298
Branch: REC
Department: I ECE AF
Batch: 2028
Degree: B.E - ECE

Scan to verify results

NeoColab_REC_CS23231_DATA STRUCTURES

REC_DS using C_Week 2_COD_Question 4

Attempt : 1
Total Mark : 10
Marks Obtained : 10

Section 1 : Coding

1. Problem Statement

Ravi is developing a student registration system for a college. To efficiently store and manage the student IDs, he decides to implement a doubly linked list where each node represents a student's ID.

In this system, each student's ID is stored sequentially, and the system needs to display all registered student IDs in the order they were entered.

Implement a program that creates a doubly linked list, inserts student IDs, and displays them in the same order.

Input Format

The first line contains an integer N the number of student IDs.

The second line contains N space-separated integers representing the student IDs.

Output Format

The output should display the single line containing N space-separated integers representing the student IDs stored in the doubly linked list.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 5
10 20 30 40 50
Output: 10 20 30 40 50

Answer

```
#include<iostream>
using namespace std;
struct node{
   int data;
   struct node* prev;
   struct node* next;
};
int main(){
   int n, id;
   cin >> n;
   struct node* head = NULL;
   struct node* tail = NULL;
   for(int i=0;i<n;i++){
      cin >> id;
      struct node* newNode = new node;
      newNode->data = id;
      newNode->prev = tail;
      newNode->next = NULL;
      if(head == NULL){
         head = newNode;
         tail = newNode;
      }else{
         tail->next = newNode;
         tail = newNode;
```

```
        }
    }
    struct node* temp = head;
    while(temp != NULL){
        cout << temp->data;
        if(temp->next != NULL)
        {
            cout << " ";
        }
        temp = temp->next;
    }
    cout << endl;
    return 0;
}
```

Status : Correct **Marks : 10/10**