Rajalakshmi Engineering College

Name: Jenell S G
Email: 240701212@rajalakshmi.edu.in
Roll no: 2116240701212
Phone: 7418493255
Branch: REC
Department: I CSE AH
Batch: 2028
Degree: B.E - CSE

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

```
// You are using GCC
#include <stdio.h>
#include <stdlib.h>

// Define the node structure
struct node {
    int id;
    struct node* prev;
    struct node* next;
};

// Initialize head and tail
struct node* head = NULL;
struct node* tail = NULL;

// Insert at the end of the list
void insert(int id) {
    struct node* nnode = (struct node*)malloc(sizeof(struct node));
    nnode->id = id;
    nnode->next = NULL;

    if (head == NULL) {
        nnode->prev = NULL;
        head = nnode;
```

```
        tail = nnode;
    } else {
        tail->next = nnode;
        nnode->prev = tail;
        tail = nnode;
    }
}

// Display the list forward
void display() {
    struct node* temp = head;
    while (temp != NULL) {
        printf("%d ", temp->id);
        temp = temp->next;
    }
    printf("\n");
}

// Main function
int main() {
    int N, id;
    scanf("%d", &N);  // Number of nodes to insert

    for (int i = 0; i < N; i++) {
        scanf("%d", &id);
        insert(id);
    }

    display();
    return 0;
}
```

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