Rajalakshmi Engineering College

Name: tharunika R
Email: 241801296@rajalakshmi.edu.in
Roll no: 241801296
Phone: 6369646218
Branch: REC
Department: I AI & DS FD
Batch: 2028
Degree: B.E - AI & DS

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 <stdio.h>
#include <stdlib.h>

// Node structure for the doubly linked list
struct Node {
    int id;
    struct Node* prev;
    struct Node* next;
};

// Create a new node
struct Node* createNode(int id) {
    struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));
    newNode->id = id;
    newNode->prev = NULL;
    newNode->next = NULL;
    return newNode;
}

// Append node at the end
void append(struct Node** head, struct Node** tail, int id) {
    struct Node* newNode = createNode(id);
    if (*head == NULL) {
        *head = *tail = newNode;
    } else {
        (*tail)->next = newNode;
        newNode->prev = *tail;
        *tail = newNode;
    }
}

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

int main() {
    int N;
    scanf("%d", &N);

    struct Node* head = NULL;
    struct Node* tail = NULL;

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

    display(head);
    return 0;
}
```

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