Rajalakshmi Engineering College

Name: Yadhu Nandhana R
Email: 241801321@rajalakshmi.edu.in
Roll no: 241801321
Phone: 7448879488
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>


struct Node{
    int studentID;
    struct Node* next;
    struct Node* prev;
};


struct Node* createNode ( int studentID) {
    struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));
    newNode->studentID = studentID;
    newNode->next = NULL;
    newNode->prev = NULL;
    return newNode;
}


void append (struct Node** head_ref, int studentID){
    struct Node* newNode = createNode(studentID);
    if(*head_ref == NULL){
        *head_ref = newNode;
```

```
        return ;
    }
    struct Node* last = *head_ref;
    while (last->next != NULL){
        last = last->next;
    }
    last->next = newNode;
    newNode->prev = last;
}


void printList(struct Node* head){
    struct Node* current = head;
    while (current !=NULL){
        printf("%d ",current->studentID);
        current = current->next;
    }
    printf("\n");
}


void freeList(struct Node* head){
    struct Node* current = head;
    while (current !=NULL){
        struct Node* next = current->next;
        free(current);
        current = next;
    }

}
int main(){
    int N ;
    struct Node* head = NULL;


    scanf("%d",&N);


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

```
    }

    printList(head);

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

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