

# Rajalakshmi Engineering College

Name: Mohamed Faheem A  
Email: 240801198@rajalakshmi.edu.in  
Roll no: 240801198  
Phone: 9952218147  
Branch: REC  
Department: I ECE FB  
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**

```
// You are using GCC
```

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
// Node structure for the doubly linked list
```

```
typedef struct Node {
```

```
    int data;
```

```
    struct Node* prev;
```

```
    struct Node* next;
```

```
} Node;
```

```
// Function to create a new node
```

```
Node* createNode(int data) {
```

```
    Node* newNode = (Node*)malloc(sizeof(Node));
```

```
    newNode->data = data;
```

```
    newNode->prev = NULL;
```

```
    newNode->next = NULL;
```

```
    return newNode;
```

```
}
```

```
// Function to insert a node at the end of the doubly linked list
```

```
void insertEnd(Node** head, int data) {
```

```
    Node* newNode = createNode(data);
```

```
    if (*head == NULL) {
```

```

        *head = newNode;
        return;
    }

    Node* temp = *head;
    while (temp->next != NULL)
        temp = temp->next;

    temp->next = newNode;
    newNode->prev = temp;
}

// Function to display the student IDs
void displayList(Node* head) {
    Node* temp = head;
    while (temp != NULL) {
        printf("%d ", temp->data);
        temp = temp->next;
    }
}

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

    Node* head = NULL;
    int id;

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

    displayList(head);
    return 0;
}

```

**Status :** Correct

**Marks :** 10/10