Rajalakshmi Engineering College

Name: Sam Devaraja J
Email: 240701463@rajalakshmi.edu.in
Roll no: 2116240701463
Phone: 7395954829
Branch: REC
Department: I CSE FE
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.

2116240701463

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>
struct node
{
   int data;
   struct node* next;
   struct node* prev;
};
typedef struct node Node;
void insertatend(Node* list,int e)
{
   Node*newnode=(Node*)malloc(sizeof(Node));
   Node* position;
   newnode->data=e;
   newnode->next=NULL;
   if(list->next==NULL)
   {
      newnode->prev=list;
      list->next=newnode;
   }
   else{
      position=list;
      while(position->next != NULL)
```

2116240701463

```
        {
            position=position->next;
        }
        newnode->prev=position;
        position->next=newnode;
    }
}
void traverse(Node* list){
    Node* position=list->next;
    while(position != NULL){
        printf("%d ",position->data);
        position=position->next;
    }
    printf("\n");
}
int main(){
    Node* list=(Node*)malloc(sizeof(Node));
    list->next=NULL;
    int n,e;
    scanf("%d",&n);
    for( int i=0;i<n;i++)
    {
        scanf("%d",&e);
        insertatend(list,e);
    }
    traverse(list);
    return 0;
}
```

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

2116240701463