Rajalakshmi Engineering College

Name: Yashwanth Kumar V
Email: 240701609@rajalakshmi.edu.in
Roll no: 240701609
Phone: 8015927564
Branch: REC
Department: I CSE FF
Batch: 2028
Degree: B.E - CSE

NeoColab_REC_CS23231_DATA STRUCTURES

REC_DS using C_Week 2_CY

Attempt : 1
Total Mark : 30
Marks Obtained : 30

Section 1 : Coding

1. Problem Statement

Sam is learning about two-way linked lists. He came across a problem where he had to populate a two-way linked list and print the original as well as the reverse order of the list. Assist him with a suitable program.

Input Format

The first line of input consists of an integer n, representing the number of elements in the list.

The second line consists of n space-separated integers, representing the elements.

Output Format

The first line displays the message: "List in original order:"

The second line displays the elements of the doubly linked list in the original order.

The third line displays the message: "List in reverse order:"

The fourth line displays the elements of the doubly linked list in reverse order.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 5
1 2 3 4 5
Output: List in original order:
1 2 3 4 5
List in reverse order:
5 4 3 2 1

Answer

```c
// You are using GCC
#include<stdio.h>
#include<stdlib.h>
struct node{
    struct node* prev;
    int data;
    struct node* next;
}*head=NULL;
typedef node Node;

void insert(Node**lis,int d){
    Node*newnode=(Node*)malloc(sizeof(Node));
    Node* pos=*lis;
    newnode->data=d;
    newnode->next=NULL;
    if(*lis==NULL){
        newnode->prev=NULL;
        *lis=newnode;
    }
    else{
        while(pos->next!=NULL){
```

```c
            pos=pos->next;
        }
        pos->next=newnode;
        pos->next->prev=pos;
    }
}

void rev1(Node*lis){
    Node*pos=lis;
    while(pos->next!=NULL && pos!=NULL)
        pos=pos->next;
    while(pos!=NULL){
        printf("%d ",pos->data);
        pos=pos->prev;
    }

}


void display(Node*lis){
    Node*pos=lis;
    if(lis==NULL)
        return;
    while(pos!=NULL){
        printf("%d ",pos->data);
        pos=pos->next;
    }
}

int main(){
    int n,i,e;
    scanf("%d",&n);
    for(i=0;i<n;i++){
        scanf("%d",&e);
        insert(&head,e);
    }
    printf("List in original order:\n");
    display(head);
    printf("\n");
    printf("List in reverse order:\n");
    rev1(head);
```

}

2.   Problem Statement

Ashiq is developing a ticketing system for a small amusement park. The park issues tickets to visitors in the order they arrive. However, due to a system change, the oldest ticket (first inserted) must be revoked instead of the last one.

To manage this, Ashiq decided to use a doubly linked list-based stack, where:

Pushing adds a new ticket to the top of the stack.Removing the first inserted ticket (removing from the bottom of the stack).Printing the remaining tickets from bottom to top.

Input Format

The first line consists of an integer n, representing the number of tickets issued.

The second line consists of n space-separated integers, each representing a ticket number in the order they were issued.

Output Format

The output prints space-separated integers, representing the remaining ticket numbers in the order from bottom to top.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 7
24 96 41 85 97 91 13
Output: 96 41 85 97 91 13

Answer

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

struct node{
    struct node* prev;
    int data;
    struct node* next;
}*head=NULL;
typedef struct node Node;

void insert(Node**lis,int d){
    Node*newnode=(Node*)malloc(sizeof(Node));
    Node*pos=*lis;
    newnode->data=d;
    newnode->next=NULL;
    if(*lis==NULL){
        newnode->prev=NULL;
        *lis=newnode;
    }
    else{
        while(pos->next!=NULL)
            pos=pos->next;
        pos->next=newnode;
        pos->next->prev=pos;
    }
}


void de1(Node*lis){
    Node*pos=lis;
    if(lis==NULL)
        return;
    pos=pos->next;
    while(pos!=NULL){
        printf("%d ",pos->data);
        pos=pos->next;
    }
}

int main(){
```

```
    int n,e,i;
    scanf("%d",&n);
    for (i=0;i<n;i++){
        scanf("%d",&e);
        insert(&head,e);
    }
    de1(head);
}
```

Status : Correct   *Marks : 10/10*

3. Problem Statement

Krishna needs to create a doubly linked list to store and display a
sequence of integers. Your task is to help write a program to read a list of
integers from input, store them in a doubly linked list, and then display the
list.

Input Format

The first line of input consists of an integer n, representing the number of
integers in the list.

The second line of input consists of n space-separated integers.

Output Format

The output prints a single line displaying the integers in the order they were
added to the doubly linked list, separated by spaces.

If nothing is added (i.e., the list is empty), it will display "List is empty".

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 5
1 2 3 4 5
Output: 1 2 3 4 5

Answer

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

struct node{
    struct node*prev;
    int data;
    struct node*next;
}*head=NULL;
typedef struct node Node;

void insert(Node**lis,int d){
    Node*newnode=(Node*)malloc(sizeof(Node));
    Node*pos=*lis;
    newnode->data=d;
    newnode->next=NULL;
    if(*lis==NULL){
        newnode->prev=NULL;
        *lis=newnode;
    }
    else{
        while(pos->next!=NULL)
            pos=pos->next;
        pos->next=newnode;
        pos->next->prev=pos;
    }
}

void display(Node*lis){
    Node*pos=lis;
    if(lis==NULL)
        printf("List is empty");
    else{
        while(pos!=NULL){
            printf("%d ",pos->data);
            pos=pos->next;
        }
    }
}

int main(){
```

```
    int n,e,i;
    scanf("%d",&n);
    for(i=0;i<n;i++){
        scanf("%d",&e);
        insert(&head,e);
    }
    display(head);
}
```

Status : Correct   *Marks : 10/10*