Rajalakshmi Engineering College

Name: KEERTHI PRIYA T
Email: 240701258@rajalakshmi.edu.in
Roll no: 2116240701258
Phone: 7397397221
Branch: REC
Department: I CSE FC
Batch: 2028
Degree: B.E - CSE

Scan to verify results

NeoColab_REC_CS23231_DATA STRUCTURES

REC_DS using C_Week 2_COD_Question 3

Attempt : 1
Total Mark : 10
Marks Obtained : 10

Section 1 : Coding

1. Problem Statement

Bob is tasked with developing a company's employee record management system. The system needs to maintain a list of employee records using a doubly linked list. Each employee is represented by a unique integer ID.

Help Bob to complete a program that adds employee records at the front, traverses the list, and prints the same for each addition of employees to the list.

Input Format

The first line of input consists of an integer N, representing the number of employees.

The second line consists of N space-separated integers, representing the employee IDs.

Output Format

For each employee ID, the program prints "Node Inserted" followed by the current state of the doubly linked list in the next line, with the data values of each node separated by spaces.

Refer to the sample output for formatting specifications.

Sample Test Case

```
Input: 4
101 102 103 104
Output: Node Inserted
101
Node Inserted
102 101
Node Inserted
103 102 101
Node Inserted
104 103 102 101
```

Answer

```
#include <iostream>
using namespace std;

struct node {
   int info;
   struct node* prev, * next;
};

struct node* start = NULL;



struct node* createNode(int data) {
   struct node* newNode = (struct node*) malloc(sizeof(struct node));
   newNode->info = data;
   newNode->prev = NULL;
   newNode->next = NULL;
   return newNode;
}
```

```
// Function to insert node at the front
void insertFront(struct node** head, int data) {
    struct node* newNode = createNode(data);
    if (*head != NULL) {
        newNode->next = *head;
        (*head)->prev = newNode;
    }
    *head = newNode;
}

// Function to print the list
void traverse() {
    struct node* temp = start;
    while (temp != NULL) {
        printf("%d ", temp->info);
        if (temp->next !=NULL){
            printf(" ");
        }
        temp = temp->next;
    }
    printf("\n");
}


void insertAtFront ( int data)
{
    insertFront(&start,data);
    printf("Node Inserted\n");
}




int main() {
    int n, data;
    cin >> n;
    for (int i = 0; i < n; ++i) {
        cin >> data;
        insertAtFront(data);
        traverse();
    }
    return 0;
```

```
}
```

Status : Correct *Marks : 10/10*