Practical no: 4

Objective: Write a program in C to create:-

1. Singly circular linked list
2. Doubly circular linked list

Program Codes: Following is the code of this problems in C:-

1. Practical4a.c

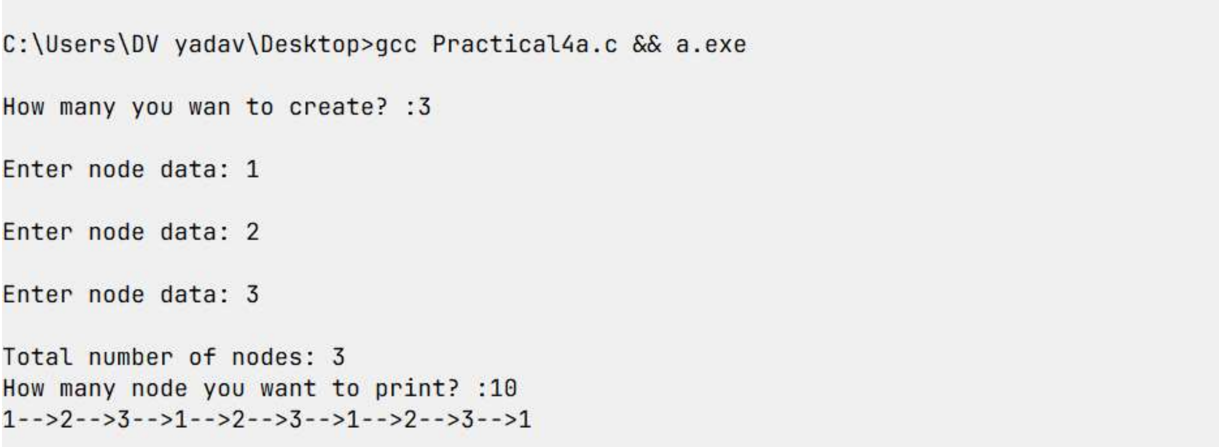
```
#include<stdio.h>
#include<stdlib.h>

typedef struct node node;

struct node{
    int data;
    node *next;
};

// head: point to first node in list
// last: points to last node in list
// count: maintains the number of nodes
node* head = NULL;
int node_count = 0;

// function declaration
void append();
void display();

void append(){

    // creating new node
    node* new_node_ptr = (node*)malloc(sizeof(node));
    new_node_ptr->next = head;  //this makes the list circular.
    printf("\nEnter node data: ");
    scanf("%d", &new_node_ptr->data);

// if list is empty: create first node
```

```
    if(head == NULL){
        head = new_node_ptr;
        node_count++;
        return;
    }

    node *temp = head;
    // traverse till last node
    for(int i = 0; i < node_count-1; i++){
        temp = temp->next;
    }
    // est. link
    temp->next = new_node_ptr;
    node_count++;
}

void display(){
    int choice;
    node *temp = head;
    printf("\nTotal number of nodes: %d"
           "\nHow many nodes do you want to print? :", node_count);
    scanf("%d", &choice);

    for(int i = 0; i<choice-1; i++){
        printf("%d-->", temp->data);
        temp = temp->next;
    }
    printf("%d\n", temp->data);
}

void main(){
    int choice;
    printf("\nHow many do you want to create? :");
    scanf("%d", &choice);

    for(int i  = 0; i <choice; i++){
        append();
    }
    display();
}
```

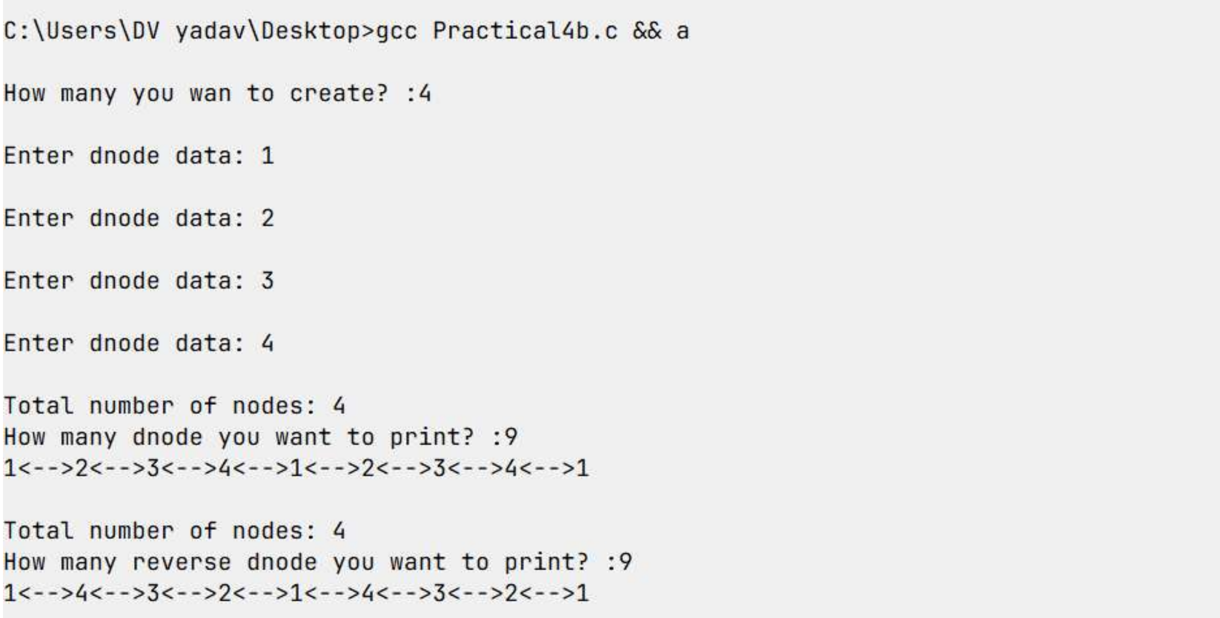
2. Practical4b.c

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

typedef struct dnode dnode;
struct dnode{
    int data;
    dnode *next;
    dnode *prev;
};

// head: point to first dnode in list
// last: points to last dnode in list
// count: maintains the number of nodes
dnode* head = NULL;
int node_count = 0;
// function declaration
void append();
void display();
void reverse_display();

void append(){
    // creating new dnode
    dnode* new_node_ptr = (dnode*)malloc(sizeof(dnode));
    new_node_ptr->next = NULL;
    new_node_ptr->prev = NULL;
    printf("\nEnter dnode data: ");
    scanf("%d", &new_node_ptr->data);
    // if list is empty: create first dnode
    if(head == NULL){
        head = new_node_ptr;
        node_count++;
        return;
    }
    dnode *temp = head;
    // traverse till last dnode
    for(int i = 0; i < node_count-1; i++){
        temp = temp->next;
    }
    // est. link
    temp->next = new_node_ptr;
    new_node_ptr→prev = temp;
    //this makes the list circular.
    new_node_ptr→next = head;
    head→prev = new_node_ptr;
    node_count++;
}

void display(){
    int choice;
    dnode *temp = head;
    printf("\nTotal number of nodes: %d"
           "\nHow many dnode you want to print? :", node_count);
    scanf("%d", &choice);
    for(int i = 0; i<choice-1; i++){
        printf("%d←-→, temp→data);
        temp = temp⟷next;
    }
    printf("%d\n", temp→data);
}

// shows the actual impl. of double circular linked list
void reverse_display(){
    int choice;
    dnode *temp = head;
    printf("\nTotal number of nodes: %d"
           "\nHow many reverse dnode you want to print? :", node_count);
    scanf("%d", &choice);
    for(int i = 0; i<choice-1; i++){
        printf("%d←—→", temp→data);
        temp = temp⟷prev;
    }
    printf("%d\n", temp→data);
}

void main(){
    int choice;
    printf("\nHow many do you want to create? :");
    scanf("%d", &choice);
    for(int i  = 0; i <choice; i++){
        append();
    }display();
    reverse_display();}
```

Output: Following is the output of the program:-

1. 

```
C:\Users\DV yadav\Desktop>gcc Practical4a.c && a.exe

How many you wan to create? :3

Enter node data: 1

Enter node data: 2

Enter node data: 3

Total number of nodes: 3
How many node you want to print? :10
1-->2-->3-->1-->2-->3-->1-->2-->3-->1
```

2. 

```
C:\Users\DV yadav\Desktop>gcc Practical4b.c && a

How many you wan to create? :4

Enter dnode data: 1

Enter dnode data: 2

Enter dnode data: 3

Enter dnode data: 4

Total number of nodes: 4
How many dnode you want to print? :9
1<-->2<-->3<-->4<-->1<-->2<-->3<-->4<-->1

Total number of nodes: 4
How many reverse dnode you want to print? :9
1<-->4<-->3<-->2<-->1<-->4<-->3<-->2<-->1
```