2. write a c program to implement circular queue using SLL.

**Code:**

#include<stdio.h>

#include<stdlib.h>

typedef struct node{

int data;

struct node \*next;

}node;

node \*createQueue(){

node \*rear=NULL,\*front=NULL;

printf("How many items you want to create in queue?");

int n,data;

scanf("%d",&n);

int i = 0;

while(i<n){

if(i==0){

printf("Enter the data: ");

scanf("%d",&data);

node \*newNode = (node \*)malloc(sizeof(node));

newNode->data = data;

newNode->next=NULL;

rear = newNode;

front = rear;

}else{

printf("Enter the data: ");

scanf("%d",&data);

node \*newNode = (node \*)malloc(sizeof(node));

newNode->data = data;

newNode->next=front;

rear->next=newNode;

rear = newNode;

}

++i;

}

return rear;

}

void display(node \*rear, node \*front){

if(!rear){

printf("\nNOTHING TO DISPLAY\n");

return;

}

node \*temp = front;

if(!rear->next){

printf("%d\n",rear->data);

return;

}

if(temp!=rear){

printf("%d ",temp->data);

display(rear, temp->next);

}else{

printf("%d\n",temp->data);

}

}

void enqueue(node \*\*rear,int data){

node \*newNode = (node \*)malloc(sizeof(node));

newNode->data = data;

if(!(\*rear)){

newNode->next=NULL;

\*rear = newNode;

return;

}

node \*temp = \*rear;

if(!(\*rear)->next){

newNode->next = temp;

}else{

newNode->next = temp->next;

}

temp->next = newNode;

\*rear = newNode;

}

int dequeue(node \*rear){

int data;

if(!rear){

printf("\nNO ITEMS IN THE QUEUE\n");

return 0;

}

if(!rear->next){

data = rear->data;

free(rear);

return data;

}

node \*temp = rear->next;

data = temp->data;

rear->next = temp->next;

free(temp);

return data;

}

int main(){

node \*rear = createQueue();

display(rear,rear->next);

enqueue(&rear,99);

display(rear,rear->next);

int data = dequeue(rear);

printf("Delete item is: %d\n",data);

display(rear,rear->next);

return 0;

}

**Output:**

