**Institute of Computer Technology**

**B. Tech. Computer Science and Engineering**

**Sub: DS**

**Course Code: 2CSE302**

**Practical – 11**

**Name: Jaymin Gondaliya**

**Enrollment No: 23162171007**

**Sem - 3**

**Branch: CS**

**Class: A**

**Batch: 32**

**Problem Definition-1:** Imagine you are developing a music player application that allows users to create and manage playlists using a circular queue data structure. The playlist has a user-defined maximum length, and if a user chooses to play a song at a particular position, it will play songs in order from that position to the end and then wrap around to the beginning.

**Code:**

*#include* <stdio.h>

*#include* <stdlib.h>

struct queue

{

    int data;

    struct queue \*next;

};

struct queue \*head = NULL;

struct queue \*createQueue(int *data*);*//* **Forward declaration**

void enqueue(int *data*)

{

    struct queue \*newQueue = createQueue(*data*);

*if* (head == NULL)

    {

        head = newQueue;

*return*;

    }

    struct queue \*last = head;

*while* (last->next != NULL)

    {

        last = last->next;

    }

    last->next = newQueue;

}

struct queue \*createQueue(int *data*)

{

    struct queue \*newQueue = (struct queue \*)malloc(sizeof(struct queue));

    newQueue->data = *data*;

    newQueue->next = NULL;

*return* newQueue;

}

void dequeue()

{

*if* (head == NULL)

    {

        printf("Queue is empty\n");

*return*;

    }

    struct queue \*temp = head;

    head = head->next;

    free(temp);

}

void display()

{

    struct queue \*temp = head;

*while* (temp != NULL)

    {

        printf("%d -> ", temp->data);

        temp = temp->next;

    }

    printf("NULL\n");

}

int main()

{

    enqueue(10);

    enqueue(20);

    enqueue(30);

    enqueue(40);

    enqueue(50);

    display();

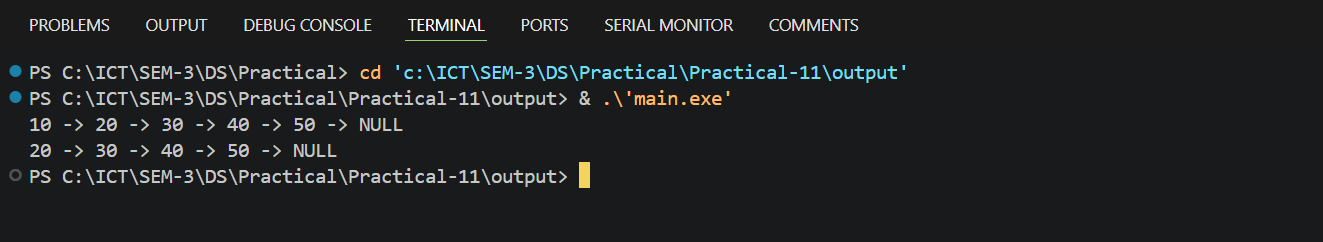
    dequeue();

    display();

*return* 0;

}

**Output:**

****