**Charotar University of Science and Technology**

**Devang Patel Institute of Advance Technology and Research**

**Department of Computer Engineering**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Student ID** | **:** | **18DCE115** | **Student Name** | **:** | **KASHYAP SHAH** |
| **Subject Code** | **:** | **CE245** | **Subject Name** | **:** | **DATA STRUCTURE AND ALGORITHM** |
| **Date of exam** | **:** | **27-05-2020** | **Semester** | **:** | **4th Semester** |

**Definition:**

Use a Circular Linked List to implement a Circular Queue. All elements are to be added by the user through the terminal. The size of the queue may not be more than 20. Write functions for: i) Adding an element ii) Removing an element iii) Searching for an element iv) Printing the list of elements in the Circular Queue.

**Solution (code):**

#include<stdio.h>

#include<conio.h>

struct node{

int data;

struct node \* next;

};

struct node \* front=0;

struct node \* rear=0;

void enqueue(int x){

struct node \* newnode;

newnode=(struct node \*)malloc(sizeof(struct node));

newnode -> data=x;

newnode -> next=0;

if(rear == 0){

front=rear=newnode;

rear->next=front;

}

else{

rear -> next=newnode;

rear=newnode;

rear -> next=front;

}

}

void dequeue(){

struct node \* temp;

temp=front;

if(front==0 && rear==0){

printf("Queue is empty.");

}

else if(front == rear){

front=rear=0;

free(temp);

}

else{

front = rear -> next;

rear -> next = front;

free(temp);

}

}

void display(){

struct node \* temp;

temp=front;

if(front==0 && rear==0){

printf("Queue is empty.");

}

else{

while(temp->next != front){

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

temp=temp->next;

}

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

}

}

void main(){

int n,a;

do{

printf("\nEnter any operation number.\n1.Enqueqe.\n2.Dequeue.\n3.Displaying the content.\n");

scanf("%d",&n);

switch(n){

case 1: printf("\nEnter any element you want to enter in circular queue : ");

scanf("%d",&a);

enqueue(a);

break;

case 2: printf("\nFirst element will be deleted.");

dequeue();

break;

case 3: printf("\nDisplaying the elements of circular queue.");

display();

break;

default: printf("\nEnter valid choice.");

exit(1);

}

}

while(1);

getch();

}

**Input and Output Screen Shot:**

