ANAGHA ACHARYA

1BM19BT005

WAP to simulate the working of a circular queue of integers using an array. Provide the following operations: a)Insert 2)Delete 3)Display

The program should print appropriate messages for queue empty and queue overflow conditions.

PROGRAM:

#include<stdio.h>

#define SIZE 5

int items[SIZE];

int front=-1,rear=-1;

int isFull(){

if((front==rear+1)||(front==0&&rear==SIZE-1))

return 1;

return 0;

}

int isEmpty(){

if(front==-1)

return 1;

return 0;

}

void enQueue(){

int element;

if(isFull())

printf("queue is full\n");

else {

printf("enter the element to be inserted:\n");

scanf("%d",&element);

if(front==-1)

front=0;

rear=(rear+1)%SIZE;

items[rear]=element;

printf("\nInserted =%d",element);

}

}

int deQueue(){

int element;

if(isEmpty())

{

printf("queue is empty");

return(-1);

}

else

{

element=items[front];

if(front==rear){

front=-1;

rear=-1;

}

else{

front=(front+1)%SIZE;

}

}

}

void display()

{

int i;

if(isEmpty())

printf("\nempty queue");

else

{

printf("\nfront=%d",front);

printf("\n items:");

for(i=front;i!=rear;i=(i+1)%SIZE){

printf("%d ",items[i]);

}

printf("%d ",items[i]);

printf("\nrear=%d\n",rear);

}

}

void main()

{

int op;

printf("\n MENU ");

printf("\n1.Insert\n2.Delete\n3.Display\n4.EXIT");

do{

printf("\nEnter your option :");

scanf("%d",&op);

switch(op)

{

case 1: enQueue();

break;

case 2:deQueue();

break;

case 3: display();

break;

case 4:exit(0);

break;

default:printf("invalid option\n");

}

} while(op!=4);

}

OUTPUT:



