**LAB ASSIGNMENT #11**

**STATEMENT:** WRITE A PROGRAM TO IMPLEMENT CIRCULAR QUEUE.

**SOURCE CODE:**

#include<stdio.h>

#include<conio.h>

#define SIZE 20

struct cqueue

{

int item[SIZE];

int rear;

int front;

};

typedef struct cqueue qu;

void insert(qu\*);

void delet(qu\*);

void display(qu\*);

void main()

{

int ch;

qu \*q;

q->rear=SIZE-1;

q->front=SIZE-1;

clrscr();

printf("Menu for program:\n");

printf("1:insert\n2:delete\n3:display\n4:exit\n");

do

{

printf("Enter your choice\n");

scanf("%d",&ch);

switch(ch)

{

case 1:

insert(q);

break;

case 2:

delet(q);

break;

case 3:

display(q);

break;

case 4:

exit(1);

break;

default:

printf("Your choice is wrong\n");

break;

}

}while(ch<5);

getch();

}

/\*\*\*\*\*\*\*\*\*\*insert function\*\*\*\*\*\*\*\*\*\*\*\*\*/

void insert(qu \*q)

{

int d;

if((q->rear+1)%SIZE==q->front)

printf("Queue is full\n");

else

{

q->rear=(q->rear+1)%SIZE;

printf ("Enter data to be inserted\n");

scanf("%d",&d);

q->item[q->rear]=d;

}

}

/\*\*\*\*\*\*\*\*\*\*delete function\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void delet(qu \*q)

{

if(q->rear==q->front)

printf("Queue is empty\n");

else

{

q->front=(q->front+1)%SIZE;

printf("Deleted item is:");

printf("%d\n",q->item[q->front]);

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*display function\*\*\*\*\*\*\*\*\*\*\*/

void display(qu \*q)

{

int i;

if(q->rear==q->front)

printf("Queue is empty\n");

else

{

printf("Items of queue are:\n");

for(i=(q->front+1)%SIZE;i!=q->rear;i=(i+1)%SIZE)

{

printf("%d\t",q->item[i]);

}

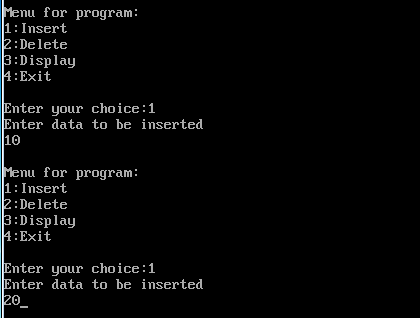
printf("%d\t",q->item[q->rear]);

}

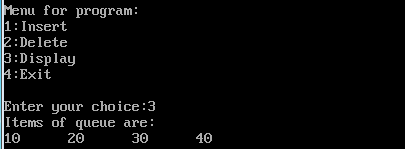
}

**OUTPUT:**

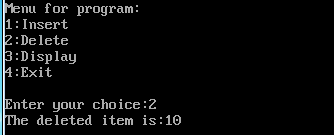
**Insert:**

****

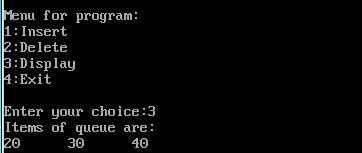
**Display:**

****

**Delete:**

****

**Display:**

****

**CONCLUSION:**

Hence, the program was successful, and the circular queue was implemented.