**Assignment No:=1.4**

**Title:=Implementation of Program based on Circular queue**

#include<iostream.h>

#include<conio.h>

class QUEUE\_

{

private:

int \*A,size,front,rear;

public:

QUEUE\_(int);

void insert(int);

int del();

void listall();

};

QUEUE\_::QUEUE\_(int par)

{

size=par;

front=rear=0;

A=new int[size+1];

}

void QUEUE\_::insert(int ele)

{

if((front==1 && rear==size)||(rear+1==front))

{

cout<<"Queue is Full"<<endl;

return;

}

if(front==0)

front=1;

if(rear==size)

rear=0;

rear++;

A[rear]=ele;

}

int QUEUE\_::del()

{

if(front==0)

{

cout<<"Queue is Empty"<<endl;

return NULL;

}

int ele=A[front];

if(front==rear)

front=rear=0;

else

{

if(front==size)

front=0;

front++;

}

return ele;

}

void QUEUE\_::listall()

{

int i;

if(front==0)

{

cout<<"Queue is Empty"<<endl;

return;

}

if(front<=rear)

{

for(i=front;i<=rear;i++)

cout<<A[i]<<"\t";

}

else

{

for(i=front;i<=size;i++)

cout<<A[i]<<"\t";

for(i=1;i<=rear;i++)

cout<<A[i]<<"\t";

}

}

void MENU()

{

int ch,size,ele;

cout<<"Enter size of queue"<<endl;

cin>>size;

QUEUE\_ obj(size);

do

{

cout<<"\n1.Insert\n2.Delete\n3.List All\n4.Exit"<<endl;

cin>>ch;

switch(ch)

{

case 1: cout<<"Enter element in queue"<<endl;

cin>>ele;

obj.insert(ele);

obj.listall();

break;

case 2:

cout<<obj.del()<<" is deleted"<<endl;

obj.listall();

break;

/\* case 3: obj.listall();

break; \*/

case 4: return;

default:cout<<"Invalid case"<<endl;

}

}while(1);

}

void main()

{

clrscr();

MENU();

getch();

}