**Exercise 6: Write a program to implement Circular Queue using arrays that performs following operations. (a) INSERT (b) DELETE (c) DISPLAY**

#include<stdlib.h>

#include<stdio.h>

#define max 5

int front=-1, rear=-1;

int CQueue[max];

void insert();

int delete();

void display();

int main()

{

int w,no;

for(;;)

{

printf("\n1. Insert");

printf("\n2. Delete");

printf("\n3. Display");

printf("\n4. EXIT");

printf("\nEnter what you want :");

scanf("%d",&w);

switch(w)

{

case 1:

insert();

break;

case 2:

no=delete();

break;

case 3:

display();

break;

case 4:

exit(1);

default:

printf("\nInvalid Choice !!\n");

}

}

}

void insert()

{

int no;

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

{

printf("\nCircular Queue Is Full !\n");

return;

}

printf("\nEnter a number to Insert :");

scanf("%d",&no);

if(front==-1)

front=front+1;

if(rear==max-1)

rear=0;

else rear=rear+1;

CQueue[rear]=no;

}

int delete()

{

int e;

if(front==-1)

{

printf("\nThe Circular Queue is Empty !!\n");

}

e=CQueue[front];

if(front==max-1)

front=0;

else if(front==rear)

{

front=-1;

rear=-1;

}

else front=front+1;

printf("\n%d was deleted !\n",e);

return e;

}

void display()

{

int i;

if(front==-1)

{

printf("\nThe Circular Queue is Empty ! Nothing To Display !!\n");

return;

}

i=front;

if(front<=rear)

{

printf("\n\n");

while(i<=rear)

printf("%d ",CQueue[i++]);

printf("\n");

}

else

{

printf("\n\n");

while(i<=max-1)

printf("%d ",CQueue[i++]) ;

i=0;

while(i<=rear)

printf("%d ",CQueue[i++]);

printf("\n");

}

}