CIRCULAR QUEUE:

#include <stdio.h>

#include <stdlib.h>

#define size 50

int Q[size];

int rear=-1;

int front=-1;

int IsFull()

{

if(front==(rear+1)%size)

{

return 0;

}

else

{

return -1;

}

}

int IsEmpty()

{

if(front==-1&&rear==-1)

{

return 0;

}

else

{

return -1;

}

}

void Enqueue(int x)

{

int item;

if(IsFull()==0)

{

printf("Queue Overflow\n");

}

else

{

if(IsEmpty()==0)

{

front=0;

rear=0;

}

else

{

rear=(rear+1)%size;

}

Q[rear]=x;

}

}

int Dequeue()

{

int x;

if(IsEmpty()==0)

{

printf("Queue underflow\n");

}

else

{

if(front==rear)

{

x=Q[front];

front=-1;

rear=-1;

}

else

{

x=Q[front];

front=(front+1)%size;

}

return x;

}

}

void Display()

{

int i;

if(IsEmpty()==0)

{

printf("Queue is empty\n");

}

else

{

printf("Queue elements:\n");

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

{

printf("%d\n",Q[i]);

}

printf("%d \n",Q[i]);

}

}

void main()

{

int choice,x,b;

while(1)

{

printf("\t\t1.Enqueue\t 2. Dequeue\t 3.Display\t 4.Exit\n");

printf("Enter your choice\n");

scanf("%d",&choice);

switch(choice)

{

case 1:

printf("Enter the number to be inserted into the queue\n");

scanf("%d",&x);

Enqueue(x);

break;

case 2:

b=Dequeue();

printf("%d was removed from the queue\n",b);

break;

case 3:

Display();

break;

case 4:

exit(1);

default:

printf("Invalid input\n");

}

}

}

