CIRCULAR QUEUE

#include<stdio.h>

int q[10],front=-1,rear=-1,MAX,choice,item,s;

int insert(int,int);

int deletion();

int display();

int search();

void main()

{

printf("enter the size of the queue<10\n");

scanf("%d",&MAX);

do

{

printf("1.ADD\n2.Delete\n3.display\n4.search\n5.exit\n");

scanf("%d",&choice);

switch(choice)

{

case 1 :

printf("Input the element for insertion in queue : \n");

scanf("%d", &item);

insert(item,MAX);

break;

case 2 :

deletion();

break;

case 3:

display();

break;

case 4:

printf("enter the item to be searched for\n");

scanf("%d",&s);

search(s);

break;

case 5:

break;

default:

printf("Wrong choice\n");

}

}while(choice!=5);

}

int insert(int item,int MAX)

{

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

{

printf("Queue Overflow \n");

return;

}

if(front == -1)

{

front = 0;

rear = 0;

}

else

{

if(rear == MAX-1)

rear = 0;

else

rear = rear+1;

}

q[rear] = item ;

}

int deletion()

{

if(front == -1)

{

printf("Queue Underflow\n");

return ;

}

printf("Element deleted from queue is : %d\n",q[front]);

if(front == rear)

{

front = -1;

rear=-1;

}

else

{

if(front == MAX-1)

front = 0;

else

front = front+1;

}

}

int display()

{

int f = front,r = rear;

if(front == -1)

{

printf("Queue is empty\n");

return;

}

printf("Queue elements :\n");

if( f <= r )

while(f <= r)

{

printf("%d ",q[f]);

f++;

}

else

{

while(f <= MAX-1)

{

printf("%d ",q[f]);

f++;

}

f = 0;

while(f <= r)

{

printf("%d ",q[f]);

f++;

}

}

printf("\n");

}

int search(int s)

{

int i;

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

{

if(s==q[i])

{

printf("element found at %d position \n",i+1);

}

}

}





