cir.c > que_size

```c
#include<stdio.h>
#include<stdlib.h>
#define que_size 3
int item,front=0,rear=-1,q[que_size],count=0;
void insertrear()
{
    if(count==que_size)
    {
        printf("queue overflow");
        return;
    }
    rear=(rear+1)%que_size;
    q[rear]=item;
    count++;
}
int deletefront()
{
    if(count==0) return -1;
    item = q[front];
    front=(front+1)%que_size;
    count=count-1;
    return item;
}
void displayq()
{
    int i,f;
    if(count==0)
    {
        printf("queue is empty");
        return;
    }
    f=front;
    printf("contents of queue \n");
    for(i=1;i<=count;i++)
    {
        printf("%d\n",q[f]);
        f=(f+1)%que_size;
    }
```

dequeue.c | cir.c | mulpri.c | despri2.c

cir.c > que_size

```c
    }
    f=front;
    printf("contents of queue \n");
    for(i=1;i<=count;i++)
    {
        printf("%d\n",q[f]);
        f=(f+1)%que_size;
    }
}
void main()
{
    int choice;
    for(;;)
    {
        printf("\n1.Insert rear \n2.Delete front \n3.Display \n4.exit \n ");
        printf("Enter the choice :\n ");
        scanf("%d",&choice);
        switch(choice)
        {
            case 1:printf("Enter the item to be inserted :\n");
                   scanf("%d",&item);
                   insertrear();
                   break;
            case 2:item=deletefront();
                   if(item==-1)
                   printf("queue is empty\n");
                   else
                   printf("item deleted is %d \n",item);
                   break;
            case 3:displayq();
                   break;
            default:exit(0);
        }
    }
    return ;

}
```

cir.c > que_size

```
1   #include<stdio.h>
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code

```
1.Insert rear
2.Delete front
3.Display
4.exit
 Enter the choice :
 1
Enter the item to be inserted :
10

1.Insert rear
2.Delete front
3.Display
4.exit
 Enter the choice :
 1
Enter the item to be inserted :
20

1.Insert rear
2.Delete front
3.Display
4.exit
 Enter the choice :
 1
Enter the item to be inserted :
30

1.Insert rear
2.Delete front
3.Display
4.exit
 Enter the choice :
 1
Enter the item to be inserted :
40
queue overflow
1.Insert rear
2.Delete front
```

C cir.c > que_size

```
1   #include<stdio.h>
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code

```
4.exit
 Enter the choice :
 1
Enter the item to be inserted :
40
queue overflow
1.Insert rear
2.Delete front
3.Display
4.exit
 Enter the choice :
 2
item deleted is 10

1.Insert rear
2.Delete front
3.Display
4.exit
 Enter the choice :
 2
item deleted is 20

1.Insert rear
2.Delete front
3.Display
4.exit
 Enter the choice :
 1
Enter the item to be inserted :
50

1.Insert rear
2.Delete front
3.Display
4.exit
 Enter the choice :
 1
Enter the item to be inserted :
60
```

cir.c > que_size

```
1   #include<stdio.h>
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code

```
4.exit
 Enter the choice :
 2
item deleted is 20

1.Insert rear
2.Delete front
3.Display
4.exit
 Enter the choice :
 1
Enter the item to be inserted :
50

1.Insert rear
2.Delete front
3.Display
4.exit
 Enter the choice :
 1
Enter the item to be inserted :
60

1.Insert rear
2.Delete front
3.Display
4.exit
 Enter the choice :
 3
contents of queue
30
50
60

1.Insert rear
2.Delete front
3.Display
4.exit
 Enter the choice :
```