19/10/20
POOJA.K
1BM19CS111

```
// Circular queue program.
#include <stdio.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 \n");
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;
```

POOJA.K
1BM19CS111

```
if (count == 0)
 {
printf ("queue is empty\n");
 return;
}
 f = front;
 printf ("Contents of queue\n");
 for (i=1; i<=count; i++)
{
 printf ("%d \n", q[f]);
 f = (f+1) % que_size;
}
}
int main()
{
 int choice;
 for(;;)
{
 printf ("\n 1: insert rear \n 2: delete front
      \n 3: display\n 4: 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 = %d \n", item);
            break;
    case 3 : displayQ();
             break;
        default: exit(0);
        }
    }
}
```

```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()
{
```

main.c

```
24  void display()
25  {
26      int i,f;
27      if(count==0)
28      {
29          printf("queue is empty");
30          return;
31      }
32      f=front;
33      printf("contents of queue \n");
34      for(i=0;i<count;i++)
35      {
36          printf("%d\n",q[f]);
37          f=(f+1)%que_size;
38      }
39  }
40  int main()
41  {
42      int choice;
43      for(;;)
44      {
45          printf("\n1.Insert rear \n2.Delete front
            \n3.Display \n4.exit \n ");
46          printf("Enter the choice : ");
47          scanf("%d",&choice);
```

main.c

```
\n3.Display \n4.exit \n ");
printf("Enter the choice : ");
scanf("%d",&choice);
switch(choice)
{
  case 1:printf("Enter the item to be inserted :");
         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 0;
}
```

```
> clang-7 -pthread -lm -o main main.c
> ./main

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 : 3
contents of queue
```



```
 Enter the choice : 3
contents of queue
10
20
30

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 : 1
Enter the item to be inserted :40

1.Insert rear
2.Delete front
3.Display
4.exit
 Enter the choice : 3
contents of queue
20
30
40
```

```
 Enter the choice : 3
contents of queue
10
20
30

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 : 1
Enter the item to be inserted :40

1.Insert rear
2.Delete front
3.Display
4.exit
 Enter the choice : 3
contents of queue
20
30
40
```

```
40

1.Insert rear
2.Delete front
3.Display
4.exit
 Enter the choice : 1
Enter the item to be inserted :50
queue overflow
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 : 2
item deleted is 30

1.Insert rear
2.Delete front
3.Display
4.exit
 Enter the choice : 2
item deleted is 40
```

```
 Enter the choice : 2
item deleted is 20

1.Insert rear
2.Delete front
3.Display
4.exit
 Enter the choice : 2
item deleted is 30

1.Insert rear
2.Delete front
3.Display
4.exit
 Enter the choice : 2
item deleted is 40

1.Insert rear
2.Delete front
3.Display
4.exit
 Enter the choice : 2
queue is empty

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