Lab - 4
Circular - Queue

```
#include<stdio.h>
#include<conio.h>
#include<process.h>
#define Que_Size 3
int
int item, front = 0, rear = -1, q[Que_Size];
 int count = 0;

void insertrear()
{
 if (count == Que_Size)
 {
  printf("Queue overflow\n");
   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 display()
{
 int i, f;
 if (count == 0)
    printf("Queue underflow \n");
     return;
 }
 f = front;
 printf (" Contents of Queue : \n");
 for (i = 1; i <= count ; i++)
{
  printf (i = 1; i <= count ; i++)
   f = (f + 1)
  printf ("%d \n", q[f]);
   f =(f+1) % que_size;
 }
}

void main ()
{
 int choice;

 for
```

P.T.O

```
for(;;)
{
printf("\n 1: insert rear \n 2: delete front \n 3: display
        \n 4: exit \n");
printf("Enter choice \n");
scanf("%d", &choice);

Switch (choice)
{
 Case 1: printf("Enter the item to be inserted
                \n");
         scanf("%d", &item);
         insert rear();
         break;
Case 2 : item = delete front();
Cas      if(item == -1){
         printf("Que is empty \n"); }
         else{
          printf("Item deleted = %d \n", item);}
         break;
Case 3 : display Q();
         break;
default : exit(0);
}
}
}
```