Lab-5

Program-5

Date- 19|10|2020

ALOK Kumar Rastogi 1BM19CS192

Algorithm. for circular queue.

```
→ To insert an element.
  IF ((Rear +1) % Max = front)
   {
      Print (overflow)'
      exit
   }

   if (front = -1 && Rear = -1)
  {  front = Rear=0
     else if ( Rear = Max-1 && front != 0)

        ~~set~~ Rear = 0
      else
          Rear = (Rear +1) % MAX
      }
      Queue [Rear] = Valco.

     exit

→ To delete an element
    if (front = -1)
      {
        Print ("Underflow")
         exit
       }
    Val = Queue [front]
     If (front = Rear)
      { front = ~~R~~EAR=-1
     else
        front = front +1
        }
    }
    exit
```

To display

```
display ( )
{.
    front_pos = front , rear_pos = rear ;
    if ( Front:==-1)
    {
        .print ( Queue is empty )
         exit
     }
     Print ( Queue Elements :)
      if ( front_pos <= rear_pos)
        while ( front_pos <= rear_pos)
         {
            Print ( queue [front_pos])
             front_pos ++ .
          }
        else .
          {
          while ( front_pos <= MAX-1)
          {  Print (  queue _arr [front pos ]).
            front pos  ++ .
            }
            front_pos = 0 ;
           while  ( front _ pos <= rear_ pos)
             {    print (  queue _ arr [front_pos]);
               front _pos ++
              }
             } exit.
```