To Overcome From the Drawback of Linear, Move to the Circular Queue

Circular Queue

- A **circular queue** has a **circular structure**.
- The last element of this queue is connected with the first element.
- [Circular Queue]

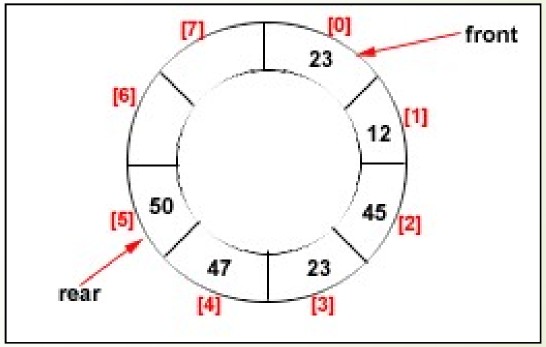
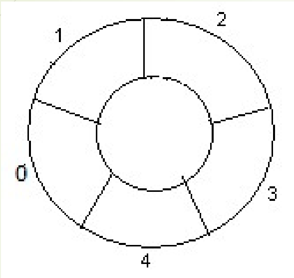


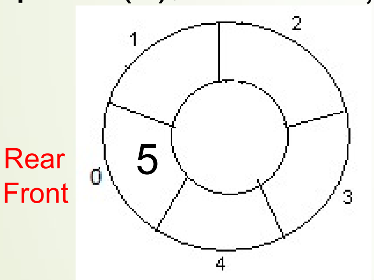
Figure: Circular Queue having Rear = 5 and Front = 0

Example: circular queue with N = 5.

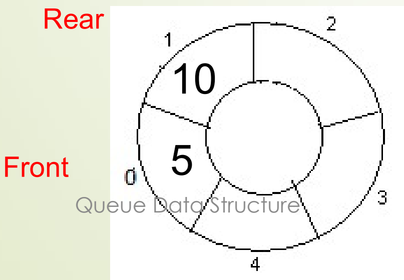
1. Initially, Rear = -1, Front = -1



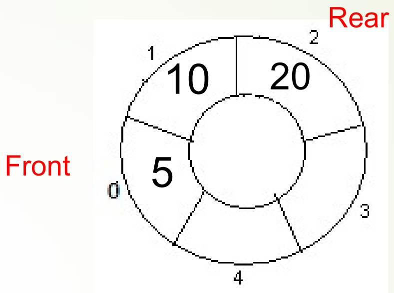
2. Enqueue(5), Rear = 0, Front = 0



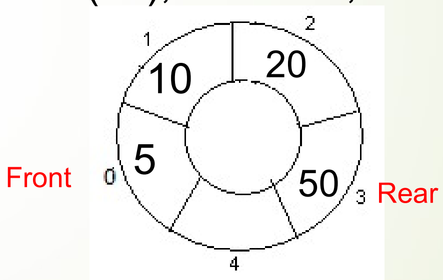
3.Enqueue(10), Rear = 1, Front = 0



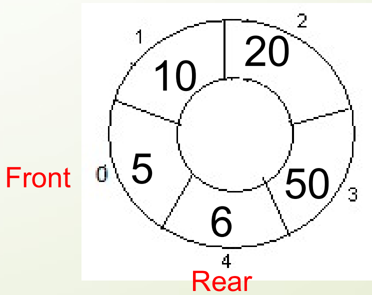
4. Enqueue(20), Rear = 2, Front = 0.



5.Enqueue(50), Rear = 3, Front = 0.

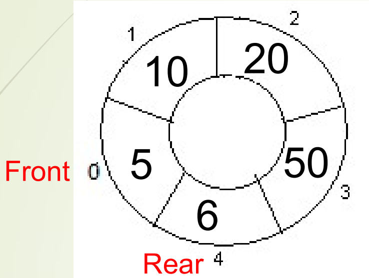


6.Enqueue(6), Rear = 4, Front = 0

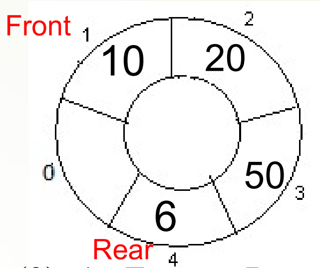


Example: circular queue with N = 5. Contd….

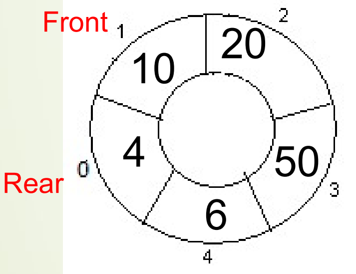
7. Enqueue(8), As Front =0 and
Rear= 4 = N-1 so **Queue overflow.**



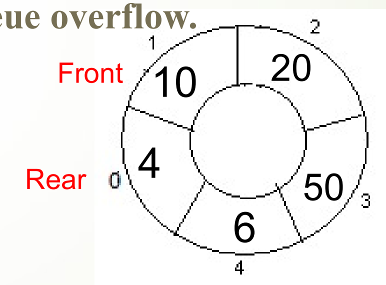
8. Dequeue() , Rear = 4, Front = 1.



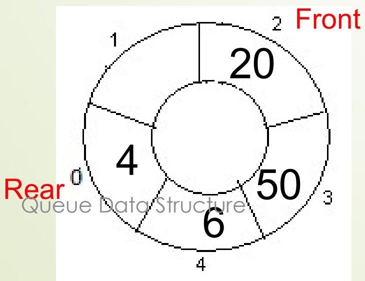
9.Enqueue(4), Rear = 0, Front = 1



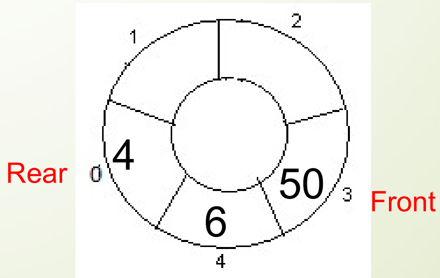
10. Enqueue(9) As Front = Rear + 1,
so **Queue overflow.**



11. Dequeue(), Rear = 0, Front = 2

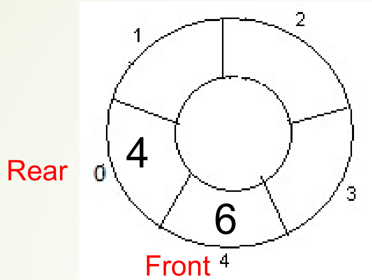


12.Dequeue(), Rear = 0, Front =3

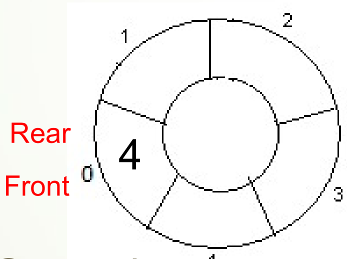


Example: circular queue with N = 5.   Contd….

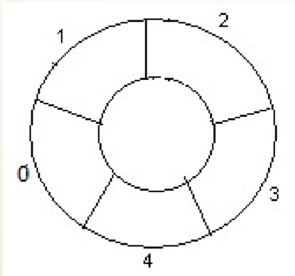
13. Dequeue, Rear = 0, Front = 4



14. Dequeue, Rear = 0, Front = 0



15. Dequeue(), as Queue is not empty and contains only one element
So updated Rear = -1, Front = -1 → Queue is Empty



Operations in Circular Queue

Algorithm IsEmpty()

step1. Start

step2. if (F=-1 and R = -1) return True ➔ Queue is Empty

step3. else return False

Step4. End

Operations in Circular Queue

Algorithm IsFull()

step1. Start

step2. if ((F=0 and R = N-1) or (F:=R+1)) ➔ Queue is Full

return True

step3. else return False

Step4. End

Operations in Circular Queue

Algorithm EnQueue(value)

- Step 1: start
- Step2: [check for queue is over flow or not] ➔ operation Validation

  If ((F=0 and R = N-1) or (F:=R+1))

  Print "queue is overflow"

  go to step 5

  else go to step 3
- Step 3: [insert item into Queue] ➔ Get the position where to insert Element

  If(R=-1) R:=F:=0

  else R:=(R+1)%N
- Step 4: QUEUE[R]:=value
- Step 5:end

Operations in Circular Queue

Algorithm DeQueue()

- Step 1: start
- Step2: [check for queue is under flow or not] ➔ operation Validation

  If (F = -1)

  Print "queue is underflow"

  go to step 5

  else go to step 3
- Step 3: [Copy item From Queue]

  X:= QUEUE[F]
- Step 4:[check condition] ➔ It is only Element of Queue or not

  If(F=R) F := R := -1 ➔ Empty Queue

  Else F:=(F+1)%N
- Step 5:end

Queue Applications

- Real life examples
  - Waiting in line
  - Waiting on hold for tech support
- Applications related to Computer Science
  - Round robin scheduling
  - An electronic mailbox is a queue
  - The ordering is chronological (by arrival time)
  - Job scheduling (FIFO Scheduling)
  - Key board buffer