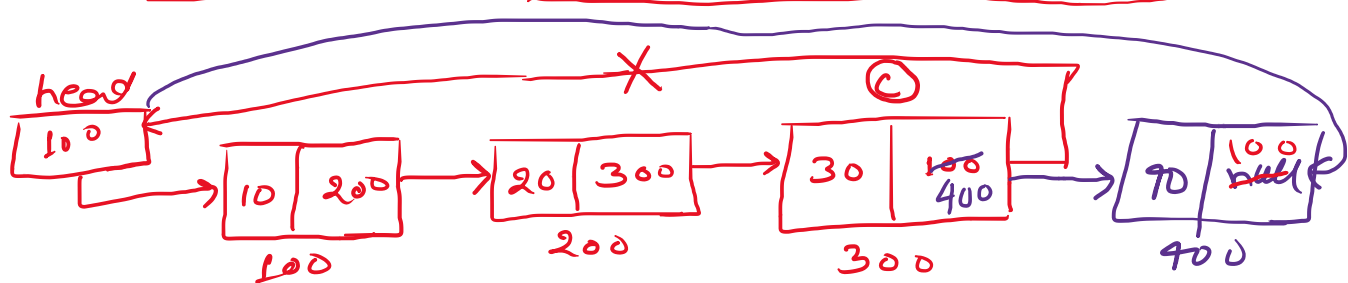


Insert node last of the circular of the list



main logic

```

n1.next = head
C.next = new n1

```

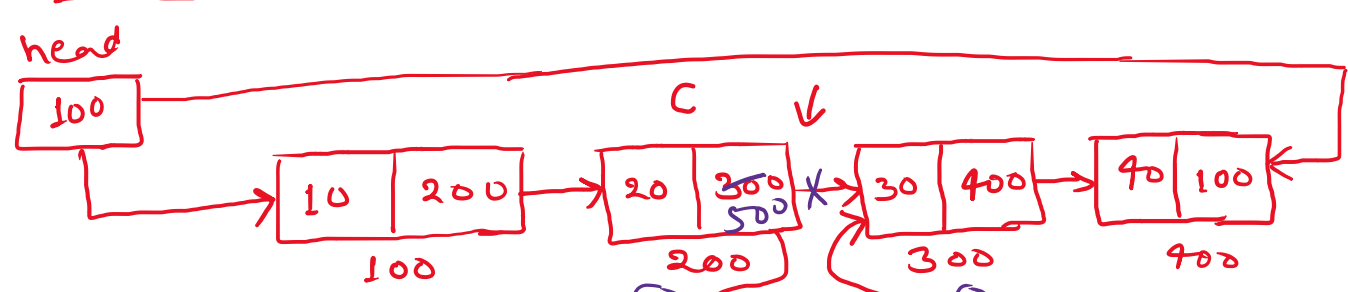
C = 100 200 300

```

void insertLast(int data){
    Node newNode = new Node(data);
    if(head != null){
        Node current = head;
        do{
            current = current.next;
        }while(current.next != head);
        current.next = newNode;
        newNode.next = head;
    }
}

```

Insert node between of the circular



h = 100, C = 100

C = h C = h.next → 200

while(C != h) { (F)

100 != 100 F

200 != 100

```

void insertBetween(int data,
int position){
    Node newNode = new Node(data);
    if(head != null){
        Node current = head;
        int ctr = 1;
        do{
            if(ctr == position){
                break;
            }
            ctr++;
            current = current.next;
        }while(current != head);
        newNode.next = current.next;
        current.next = newNode;
    }
}

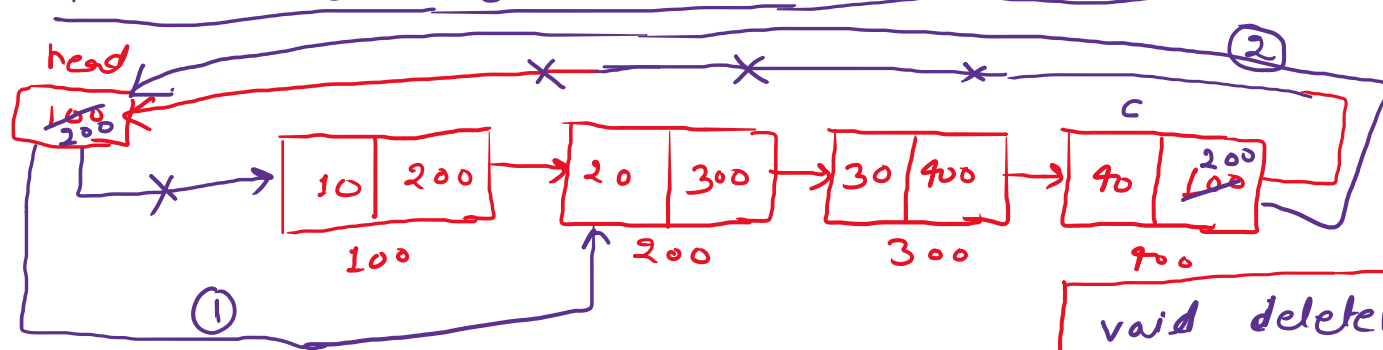
```

```

200.next = 300
newNode.next = current.next
C.next = newNode
} while(C != h)

```

Delete beginning of the circular linked list

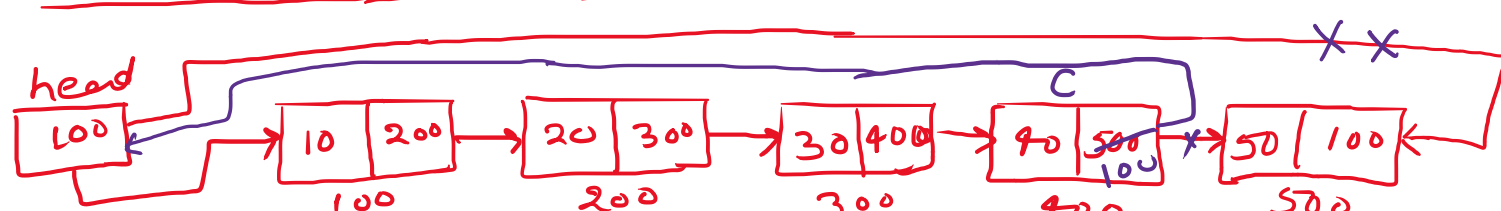


```

void deleteBeginning(){
    if(head != null){
        Node current = head;
        do{
            current = current.next;
        }while(current.next != head);
        head = head.next;
        current.next = head;
    }
}

```

Delete end of the circular linked list



```

void deleteEndNode(){
    if(head != null){
        Node current = head;
        do{
            current = current.next;
        }while(current.next.next != head);
        C.next.next = null;
        C.next = head;
    }
}

```

C.next != head

C.next.next

C = head; 100

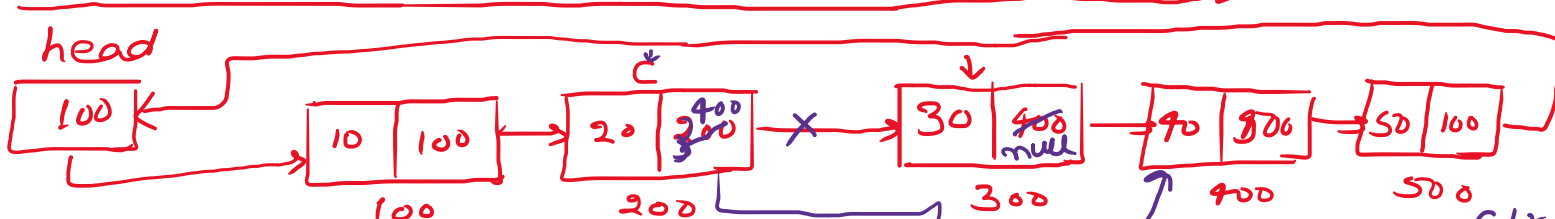
C = 200

C = 300

C.next.next = null

C.next = head;

Delete between of the circular linked list



Position = 3

```

void deleteBetween(int position){
    if(head != null){
        Node current = head;
        int ctr = 1;
        do{
            if(ctr == position){
                break;
            }
            ctr++;
            current = current.next;
        }while(current.next != head);
        current.next = current.next.next;
    }
}

```

```

ctr = 12
C = 100 200
200.next.next = null
C.next =
Node temp = current.next.next;
900 200.next
300.next
900
C = C;
C.next.next = null
C.next = temp;
temp = null;

```

17-07-2023 covered topic

① Insert into the circular linked list (last, between)

② Delete in the circular linked list (beginning, between, last)

18-07-2023 topic

① Reverse of the circular linked list

② Insert doubly circular list (beginning, between, end).