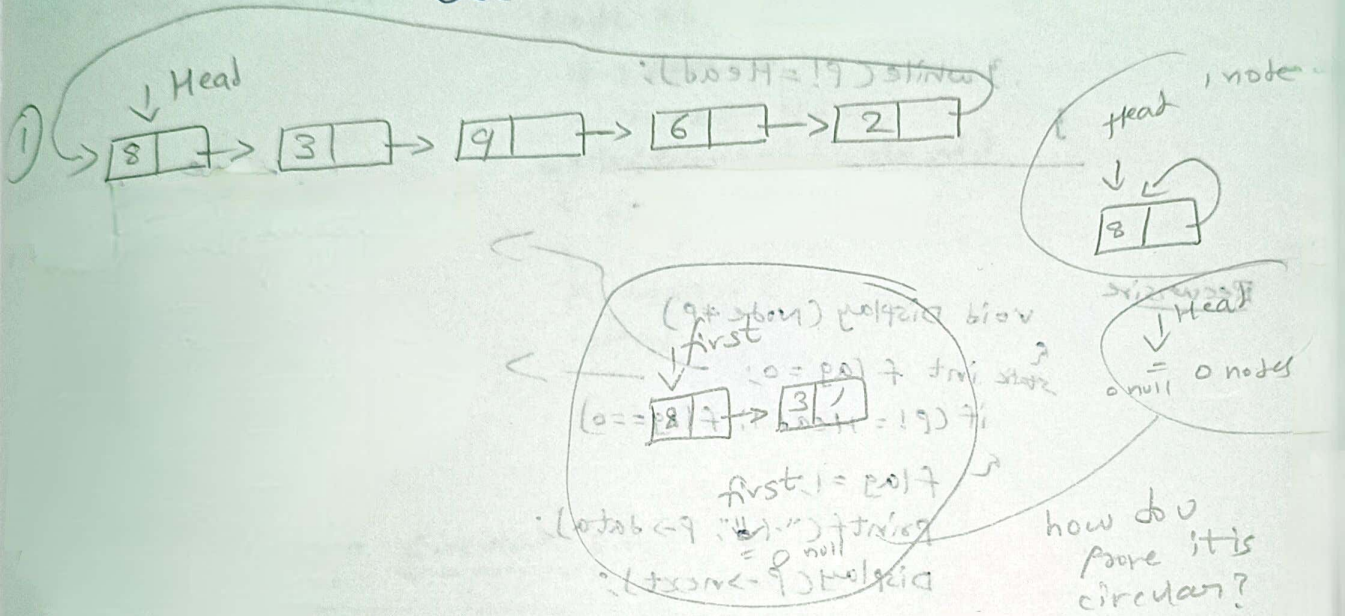
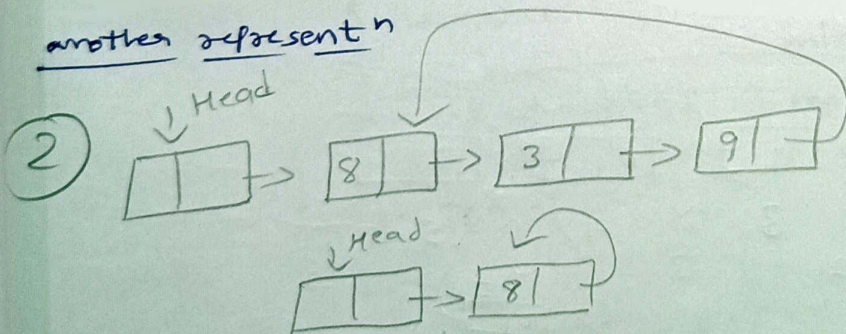


Circular LL

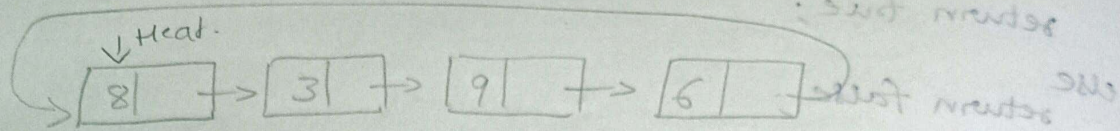


another representⁿ



Now it is empty & it is circular.

Display circular LL



```

Void Display (Node *p)
{
    do
    {
        printf("%d", p->data);
        p = p->next;
    } while (p != Head);
}
  
```

Display(Head);

Recursive

```

void Display (Node *p)
{
    static int flag = 0;
    if (p != Head || flag == 0)
    {
        flag = 1;
        printf("%d", p->data);
        Display(p->next);
    }
    flag = 0;
}
  
```

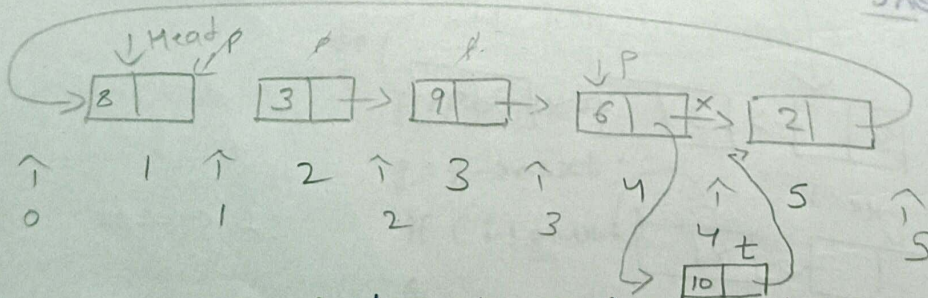
but every time in action records flag is 0, so to make it 1

This static variable is just like a global variable
(declare it as global var static inside a fn)

Display(Head);

Code pth

Inserting in a circular LL



Node *t; 1. insert before Head

Node *p = Head; 2. insert at any other position.

for (i = 0; i < pos - 1; i++)

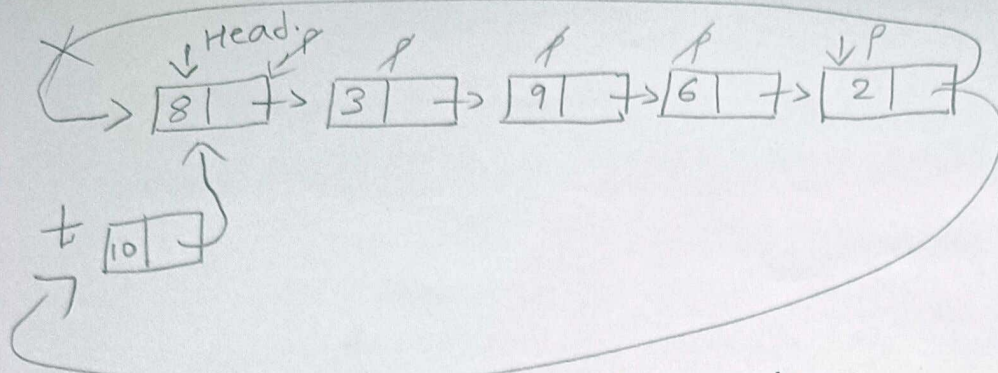
p = p->next;

t = new Node;

t->data = x;

t->next = p->next;

p->next = t;

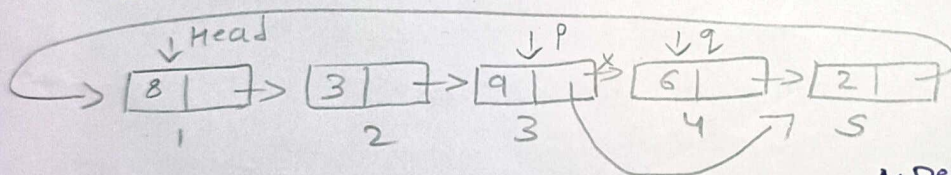


→ Head, or wish

```
Node *P = Head;
Node *t = new Node;
t->data = x;
t->next = Head;
while(P->next != Head)
{
    P = P->next;
}
P->next = t;
Head = t;
```

Code pdf ✓
in before pdf.

Deleting from Circular LL

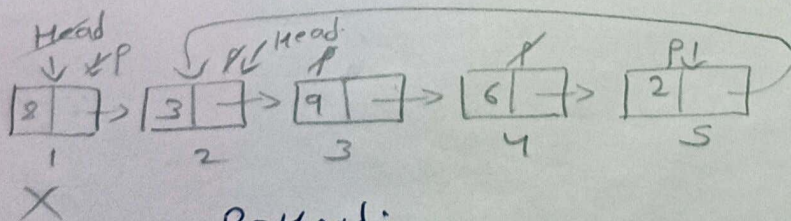


1. Deleting Head Node.
2. Deleting Node from given position.

at 4

```
P = Head;
for (i = 0; i < pos - 2; i++)
{
    P = P->next;
}
Q = P->next;
P->next = Q->next;
x = Q->data;
delete Q;
```

min $O(1)$
max $O(n)$



```
P = Head;
while(P->next != Head)
{
    P = P->next;
}
P->next = Head->next;
x = Head->data;
delete Head;
Head = P->next;
```

Code pdf ✓