# CS & IT ENGINEERING

## Data Structure

**Linked List**
**Chapter- 3**
**Lec- 06**

By- Pankaj Sharma sir

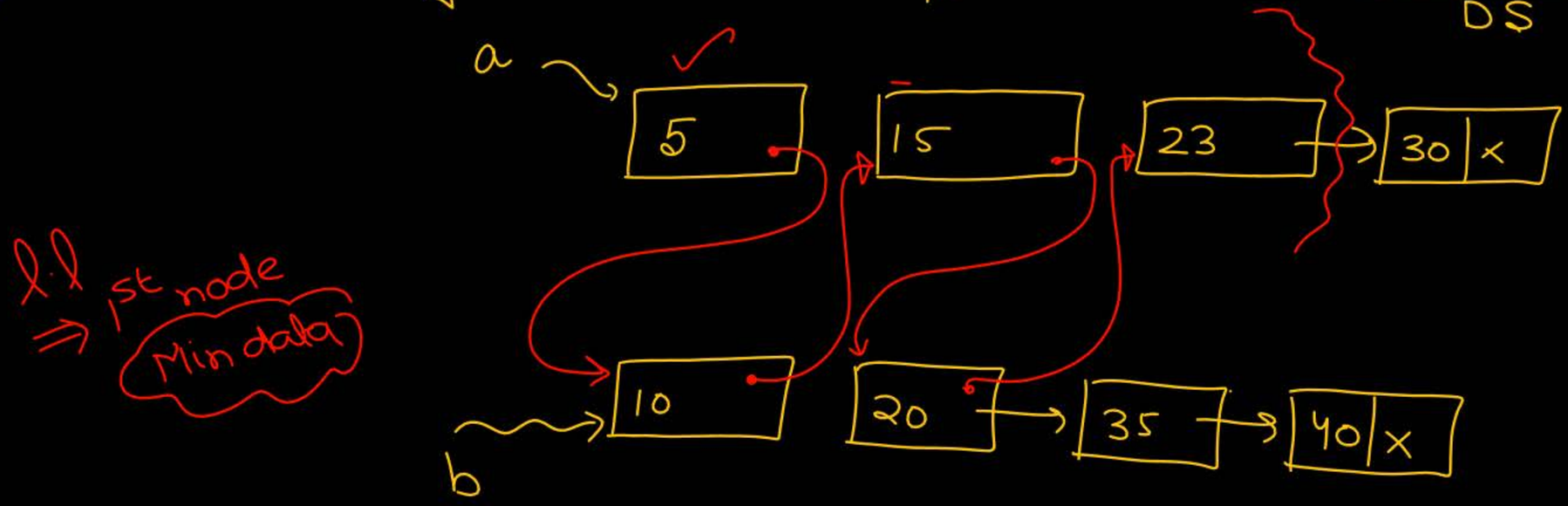

Linked List-VI

1.

# Merge 2 Sorted linked list

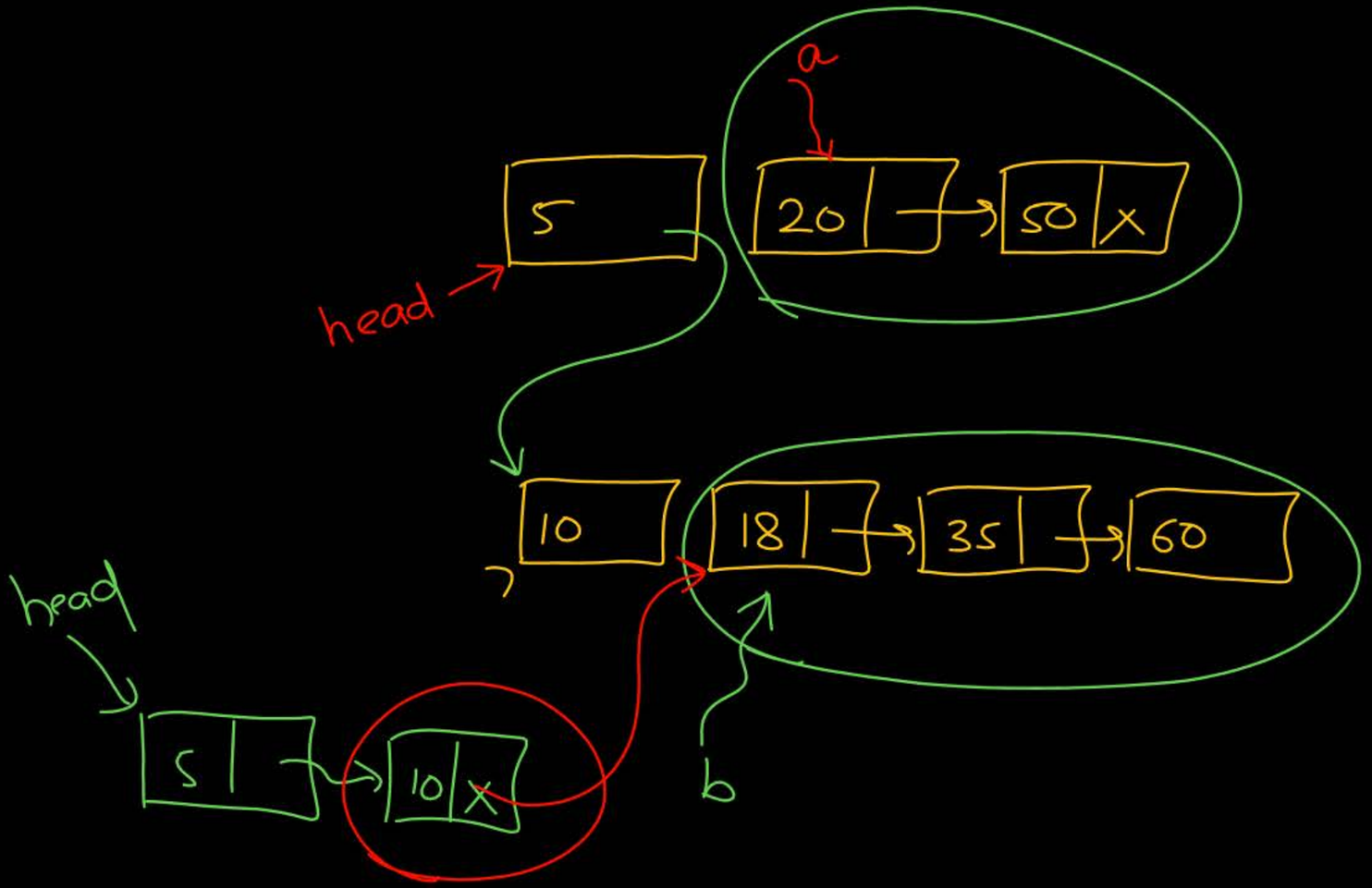
8:30 ⇒ PM

DS




head : First node in res. l.l.

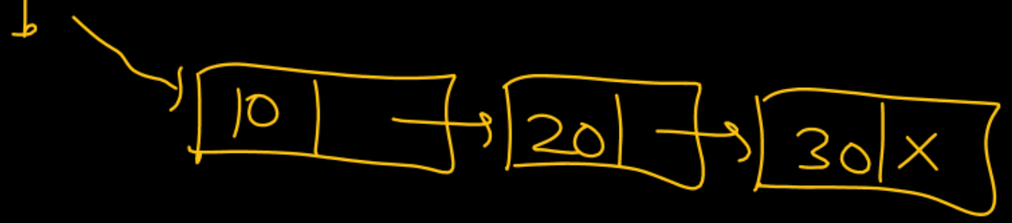
tail : Last node in res. l.l


a
5
20
50
X
head
10
18
35
60
head
5
10
X
b

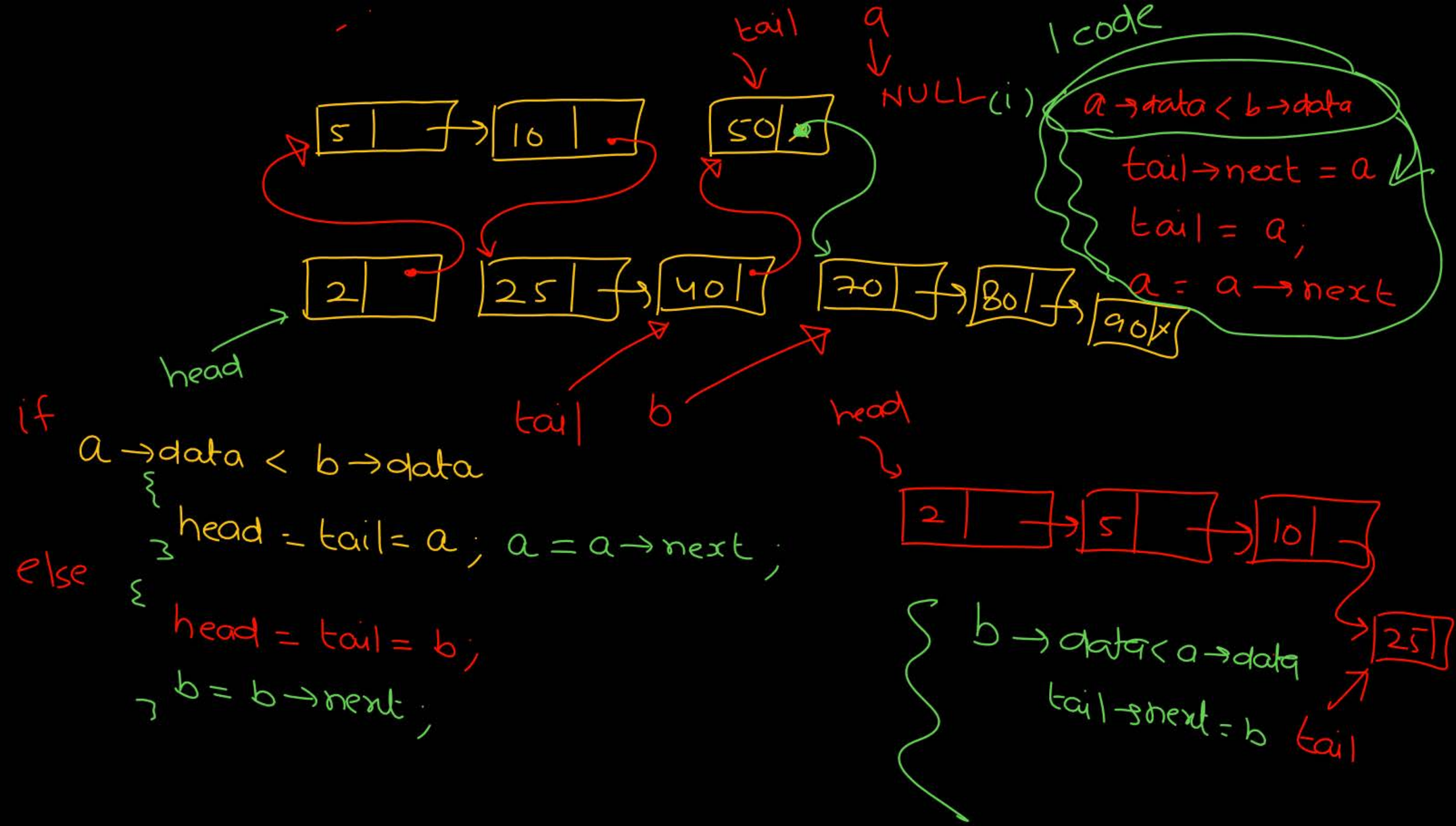
Q


o/p: return a



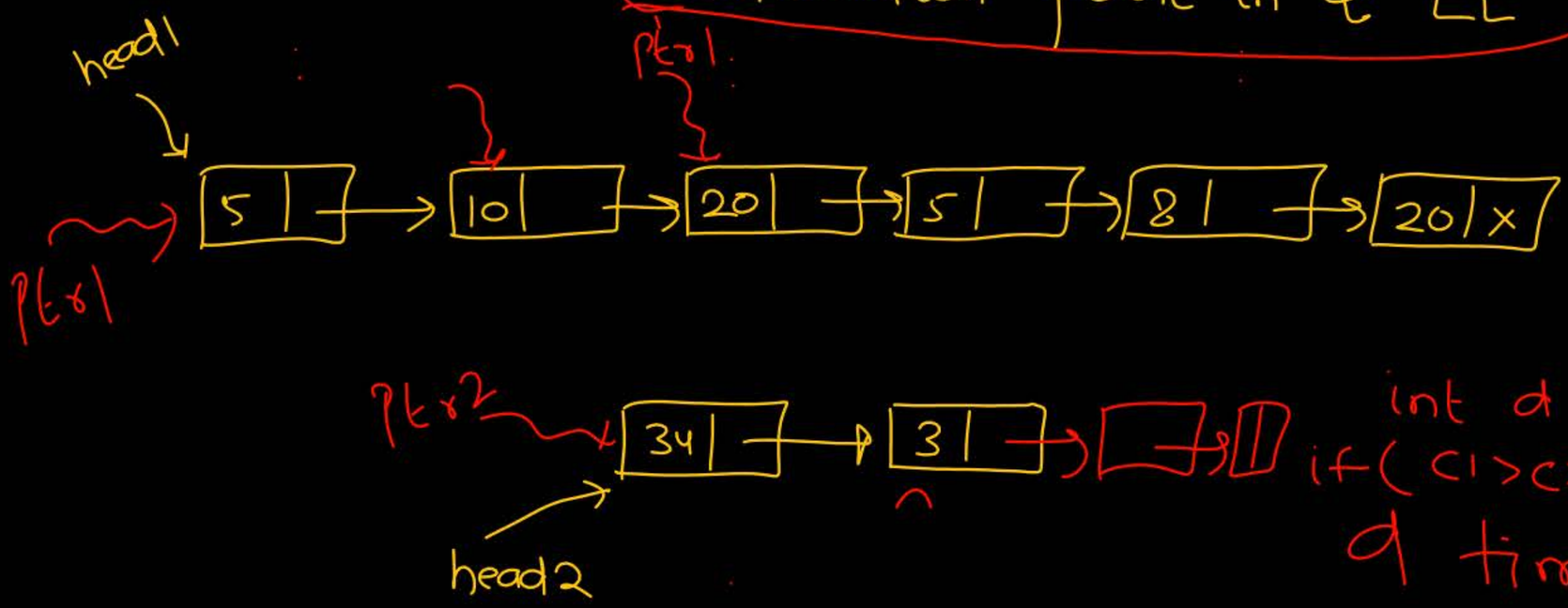



o/p return b




```
if a→data < b→data
{
    head = tail = a; a = a→next;
}
else
{
    head = tail = b;
    b = b→next;
}
```

# Intersection point in 2 LL

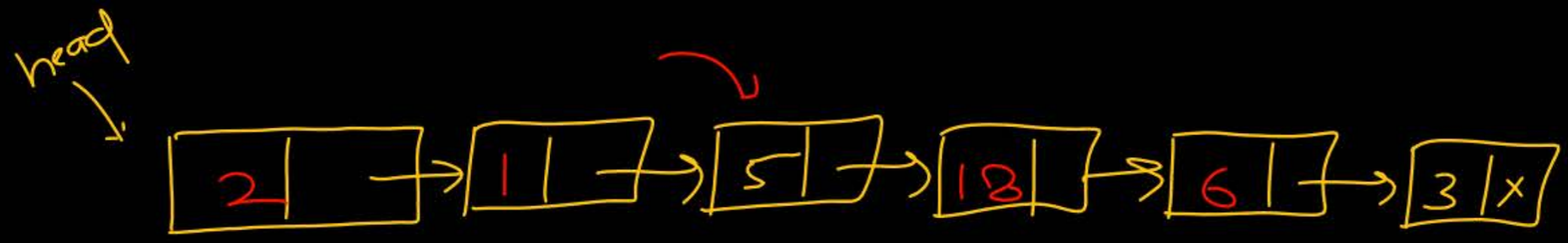



d = 2

```
int d = abs(c1 - c2)
if( c1 > c2)
    d times
        ptr1 ⇒ update
Else
    d times
        ptr2 ⇒ update
```

```
c1 = 0;
ptr1 = head1;
While (ptr1)
{
    c1++;
    ptr1 = ptr1 → next;
}
```

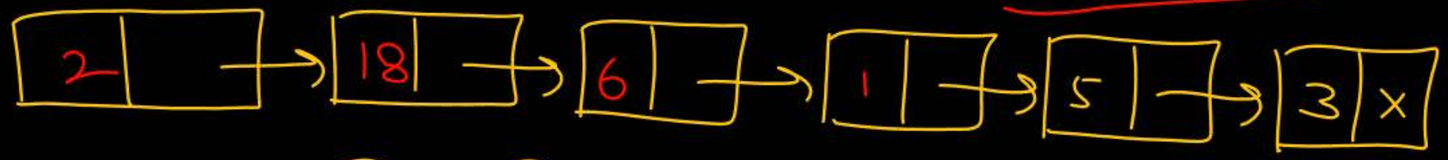
```
c2 = 0;
ptr2 = head2
while(ptr2)
{
    c2++;
    ptr2 = ptr2 → next;
}
```



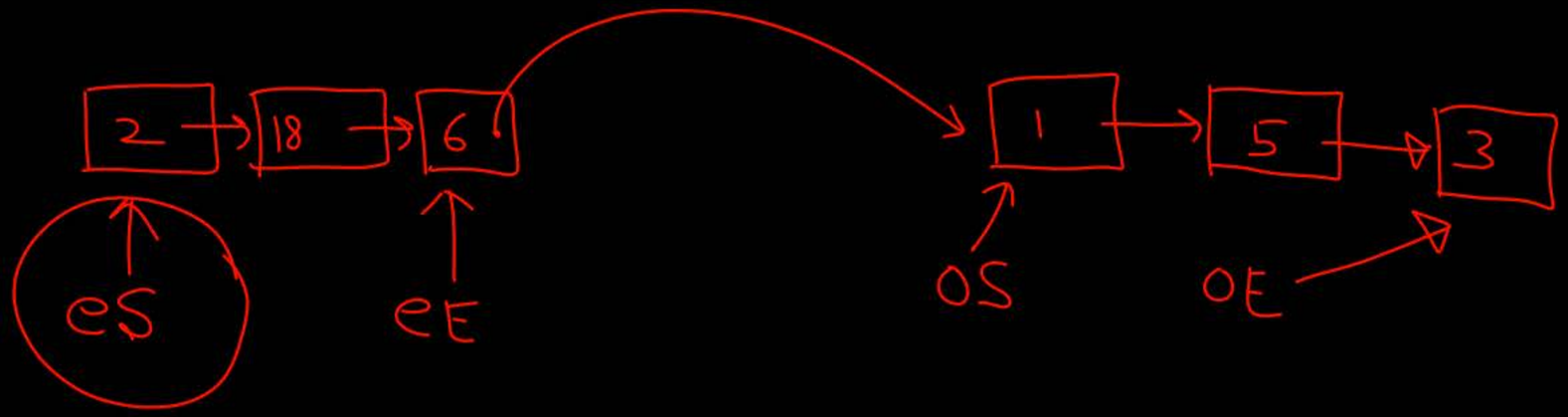

# Segregate Even & odd in a LL

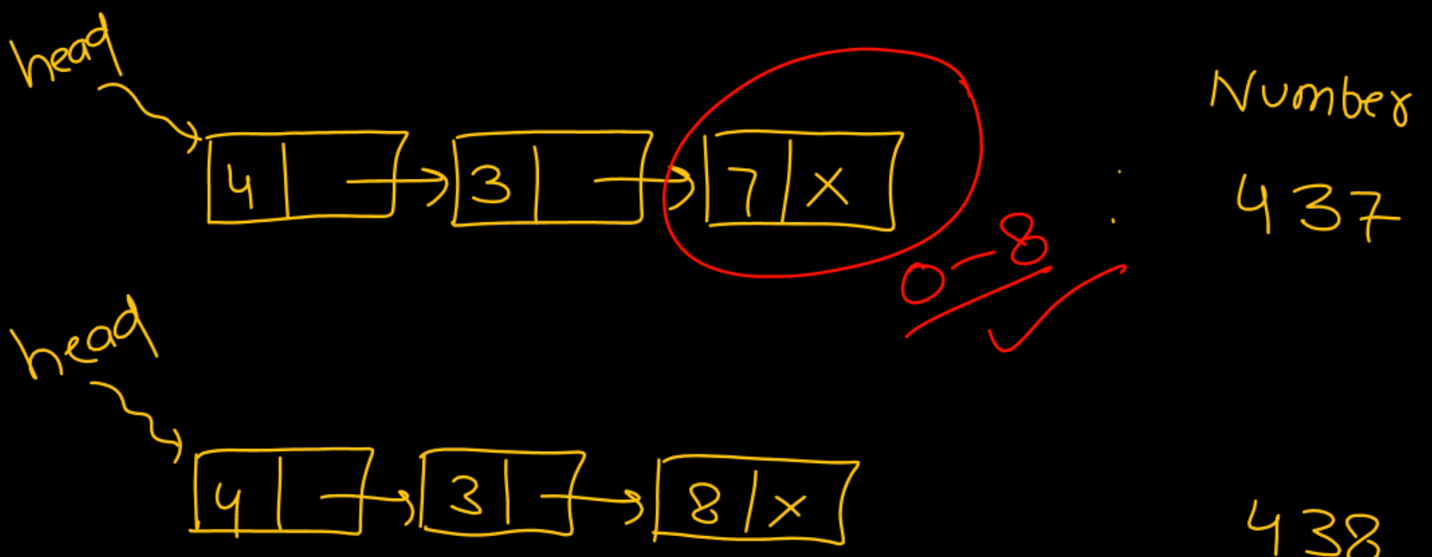
gfg Practice

O/p :




Odd



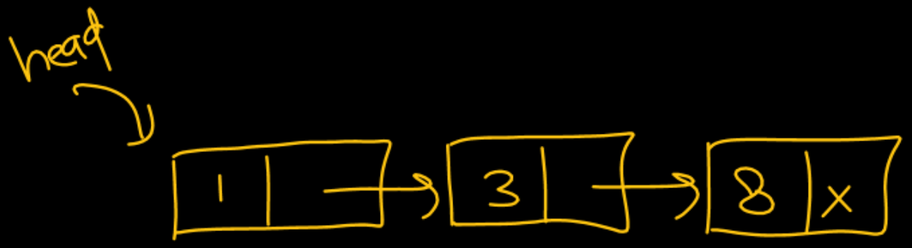


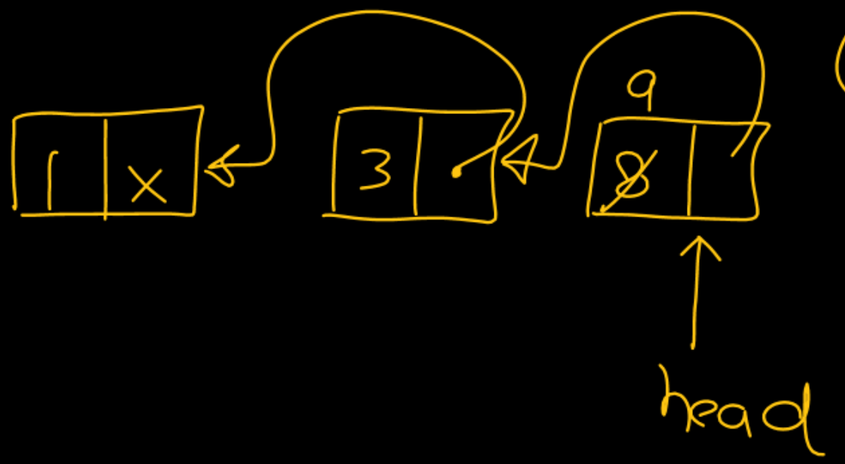

Case 1: last node ⇒ 0-8 ✓ +1 ✓


1
3
9
x


S.L.L
0


reverse

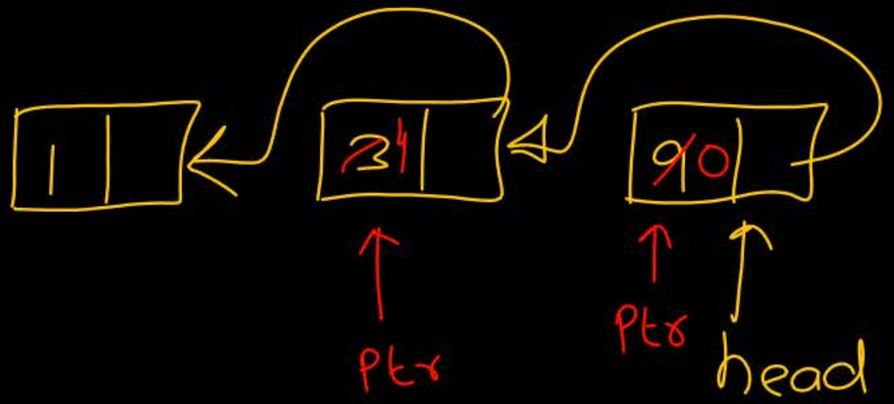


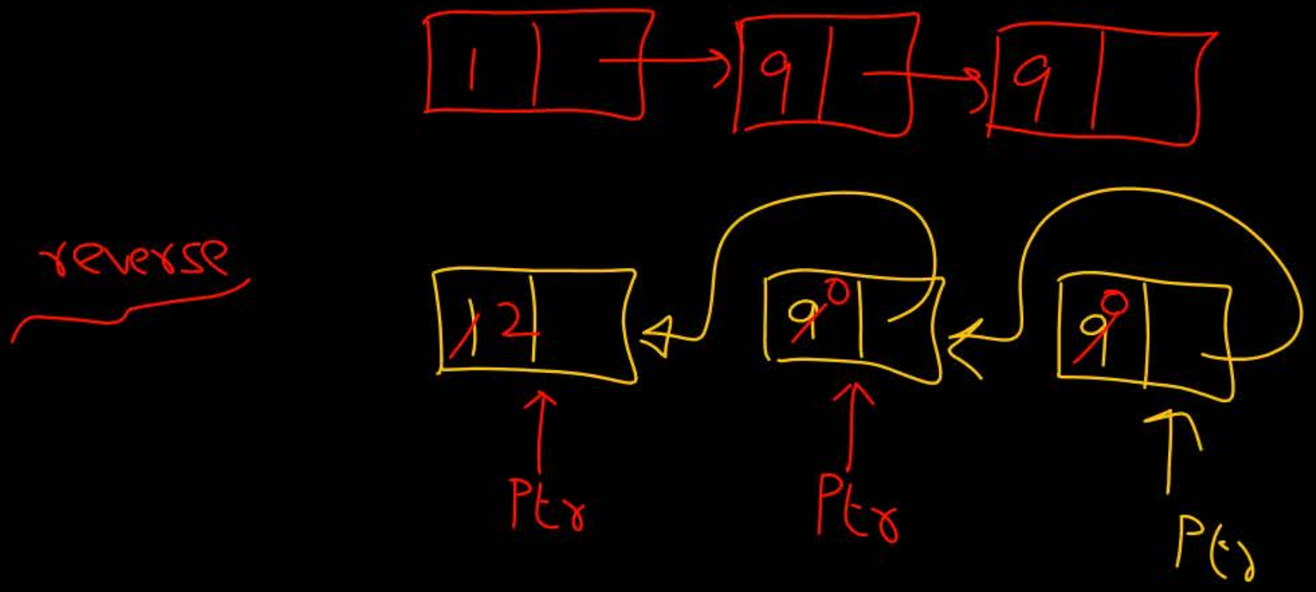



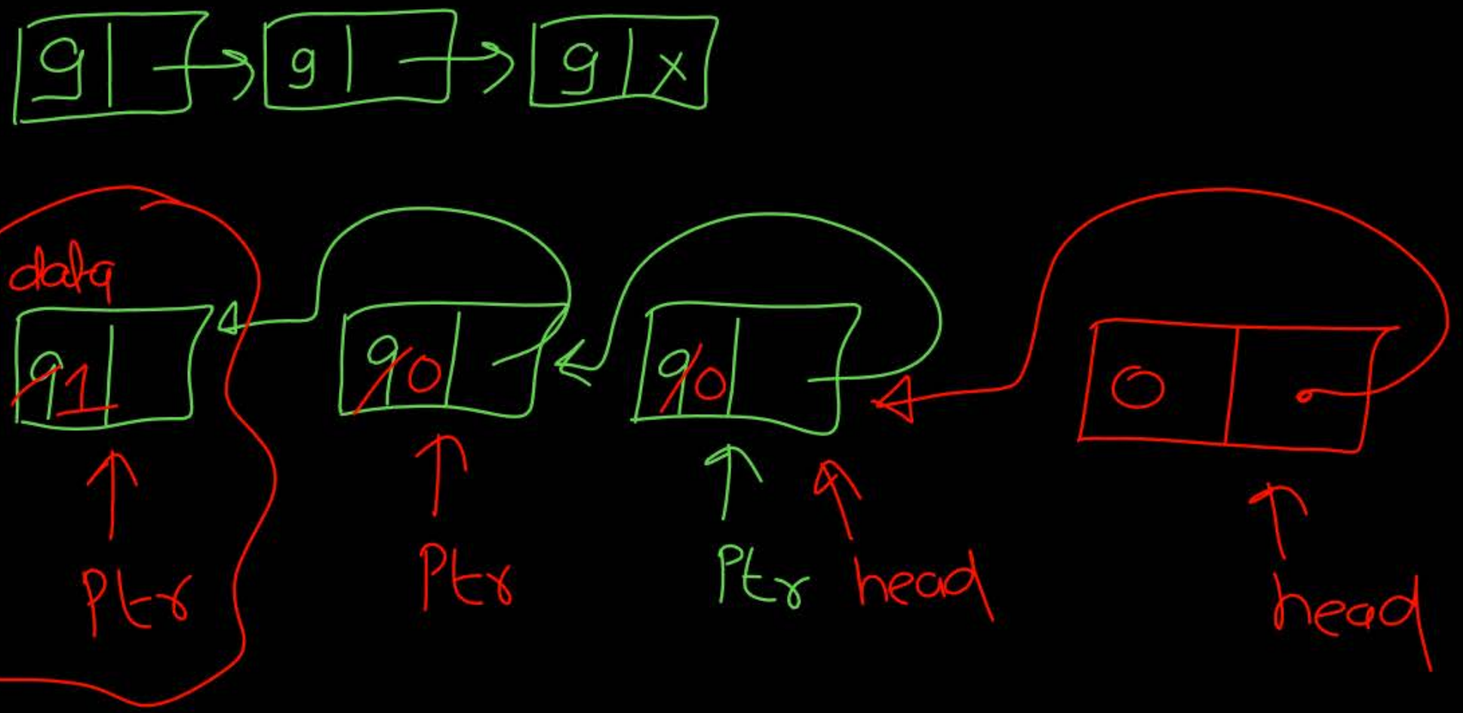

① head = reverse(head)

② if now first node 0—8 ⇓ inc by 1 and

③ reverse the LL & return head


1
34
90
Ptr
Ptr
head


1
9
9
reverse
12
90
90
Ptr
Ptr
Ptr




999

⇒ 1000

special case

Ptr → data == 9 && Ptr → next == NULL

Add 1 to Linked list

Medium - hard


THANK YOU GW
SOLDIERS !