Today's agenda

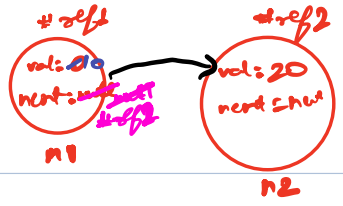
- insert in a linkedlist
- delete in a linkedlist
- Reverse the linkedlist
- Find mid
- floyd cycle.


```
int n = 10;
```

```
Public class node {
    int n;
    String s1;

}

Node temp = new Node();
```




```
class Node {
    int val;
    Node next;
    Node(int v1) {
        val = v1;
    }
}
```

```
Node n1 = new Node();
n1.val = 10;

Node n2 = new Node();
n2.val = 20;

Print(n1.next.val)

n1.next = n2;
```

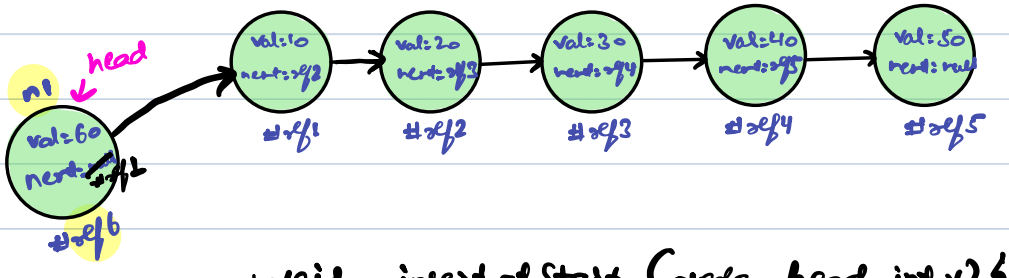
Q) insert in a linkedlist

Print (head.next.next.next.val);

u. 40

insert at head

u v=60



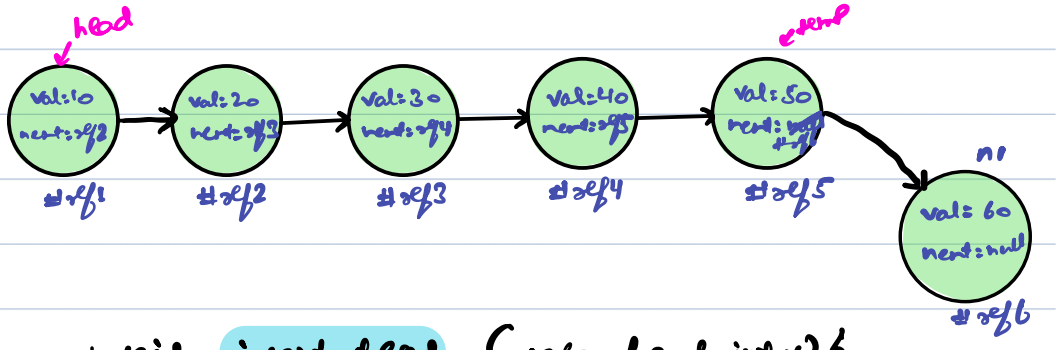

```
void insertAtStart (Node head, int v) {
    Node n1 = new Node(v);
    n1.next = head;
    head = n1;
}
```

T.C: O(1)

S.C: O(1)

insert after last node

↳ y=60



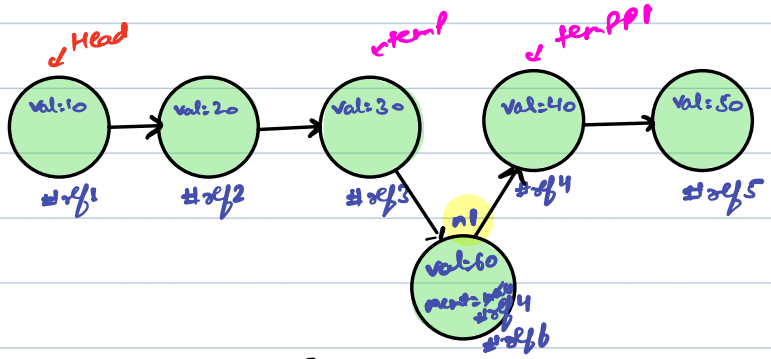

```
void insertAtEnd (Node head, int v) {
    Node n1 = new Node(v);
    Node temp = head;
    while (temp.next != null) {
        temp = temp.next;
    }
    temp.next = n1;
}
```

T.C: O(N)

S.C: O(1)

Head → null

"insert at index" → K=3, V=60




```
void insert (Node head, int v, int k) {
    Node n1 = new Node(v);
    Node temp = head;
    for (int i=1; i<=k-1; i++) {
        temp = temp.next;
    }
    Node temppp1 = temp.next;
    temp.next = n1;
    n1.next = temppp1;
}
```

T.C: O(K) = O(N)

S.C: O(1)

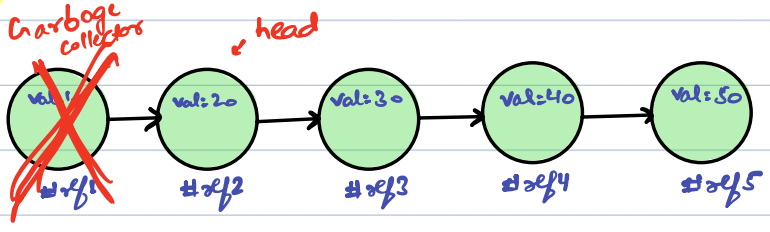
edge cases

- K==0
- head==null

Break till 9:40 PM

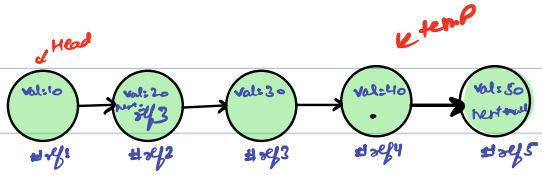
Q) Delete in a linkedlist

→ delete Head



↳ head = head.next;

→ delete last



for 2nd last node
temp.next.next=null

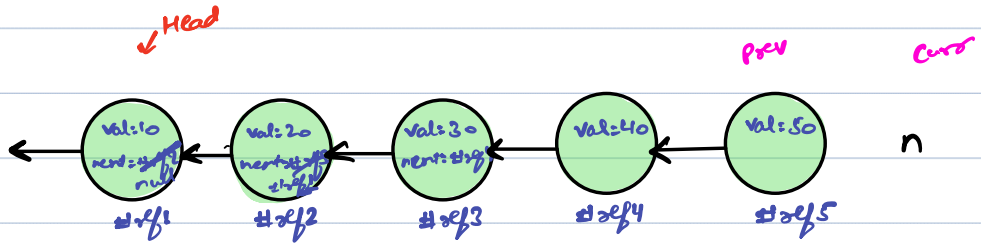
```
deleteLast ( node head){
    Node temp : head;
    while (temp.next.next !=null ) {
        temp : temp.next;
    }

    temp.next = null;
}
```

T.C: O(N)

S.C: O(1)

Q) Reverse a linkedlist



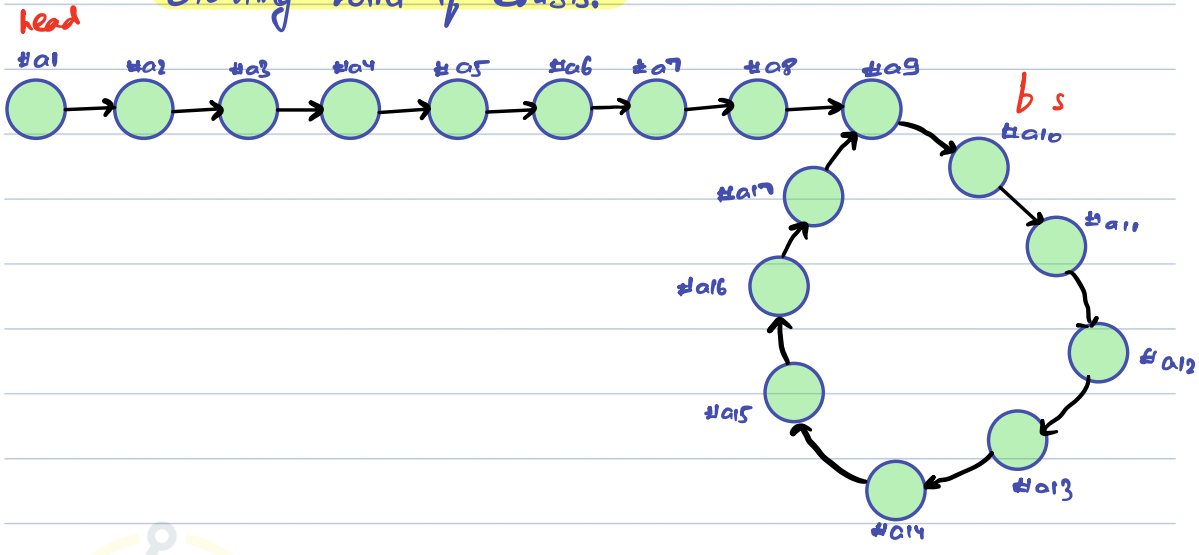

```
void reverse(Node head) {
    Node curr = head;
    Node prev = null;
    while (curr != null) {
        Node currPl = curr.next;
        curr.next = prev;
        prev = curr;
        curr = currPl;
    }
    head = prev;
}
```

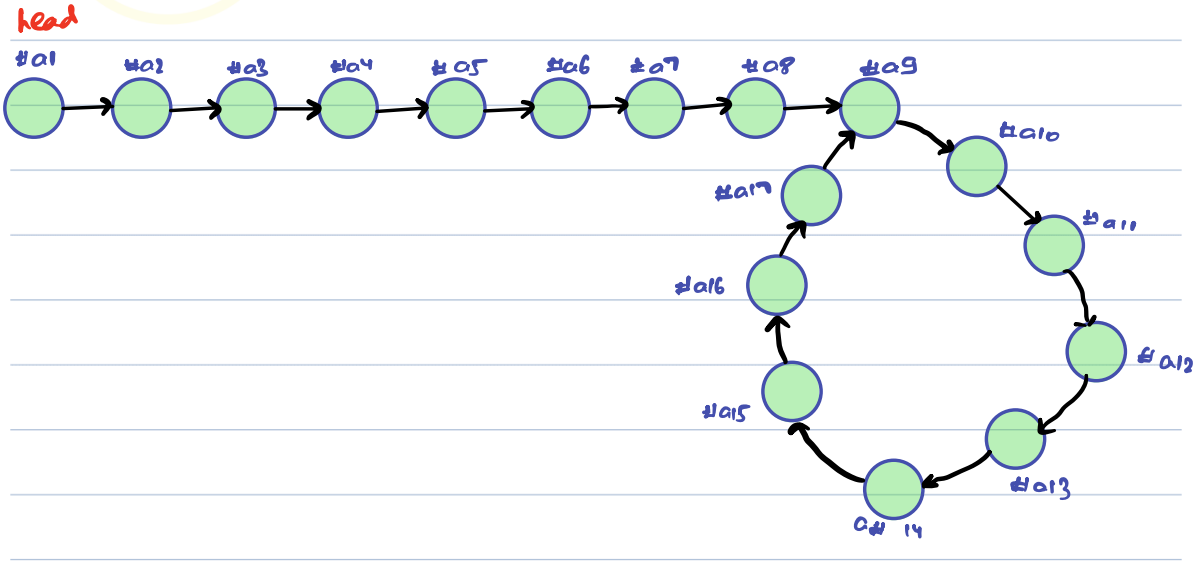
T.C : O(N)

S.C : O(1)

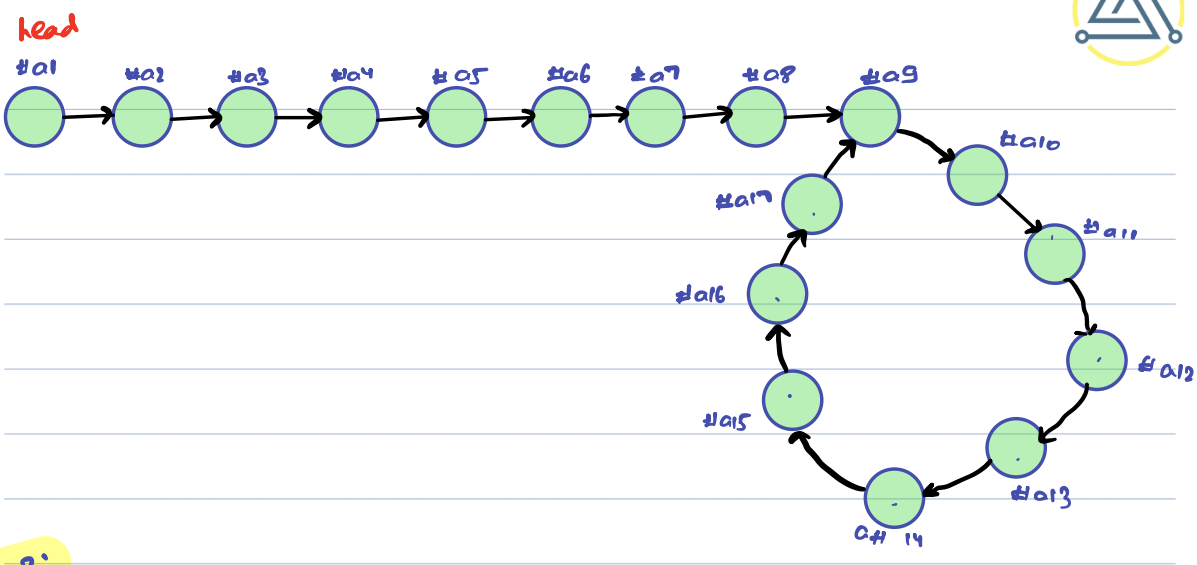
floyd cycle

Q) Given a linked list, check for cycle & return the starting point if exists.



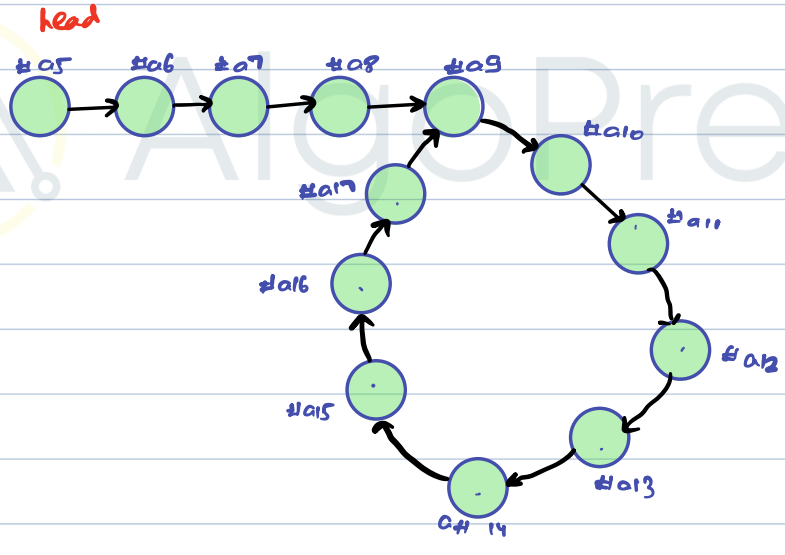


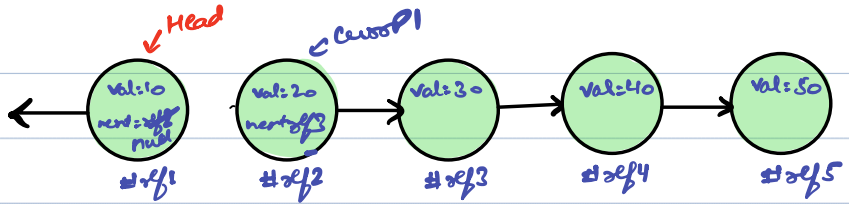
Ex1:



Ex2:

Chain: 4 Cycle: 9






```
While (    ) {
  Node CurrPl = Curr.next;
  Curr.next = Prev;
    Prev = Curr;
    Curr = CurrPl;
}
```