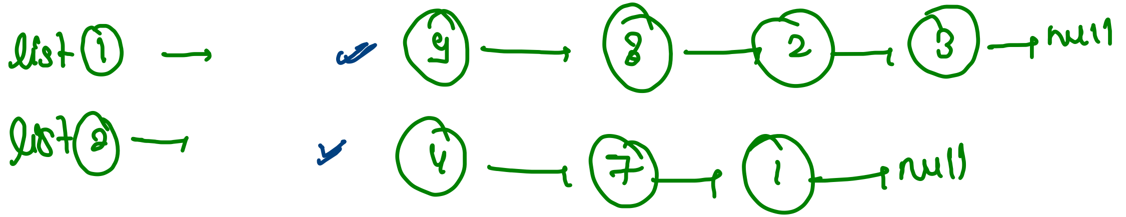
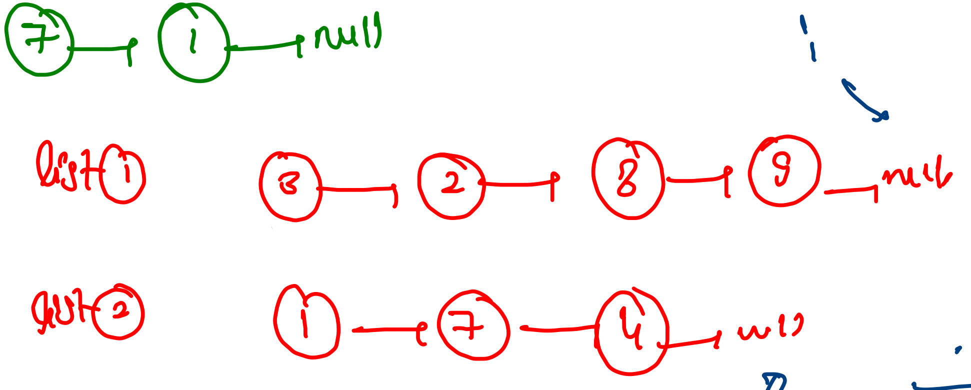
Addition of two linked list :



① → Reverse list①

② → Reverse list②



linkedlist

carry = 0

ival = 3

i = i.next;

jval = 1

j = j.next

sum = ival + jval + carry

val = sum % 10

carry = sum/10

res.add ~~list~~(val);

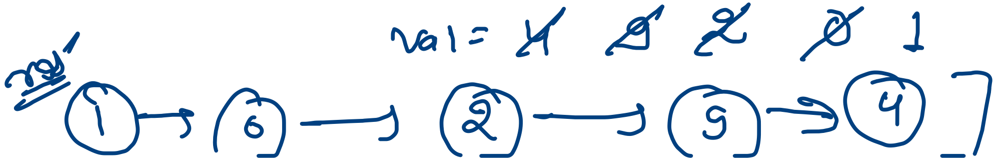
carry = ~~0~~ ~~1~~ 0

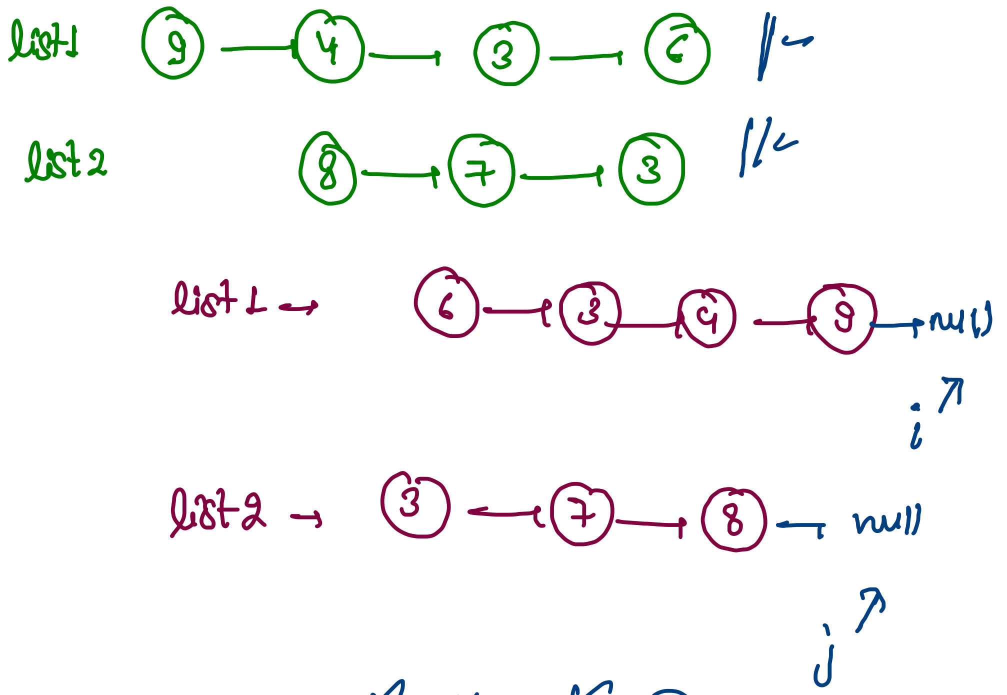
ival = ~~3~~ ~~2~~ ~~8~~ ~~9~~ 0

jval = ~~1~~ ~~7~~ ~~4~~ 0

sum = ~~4~~ ~~9~~ ~~12~~ ~~10~~ 1

val = ~~4~~ ~~9~~ ~~2~~ ~~0~~ 1





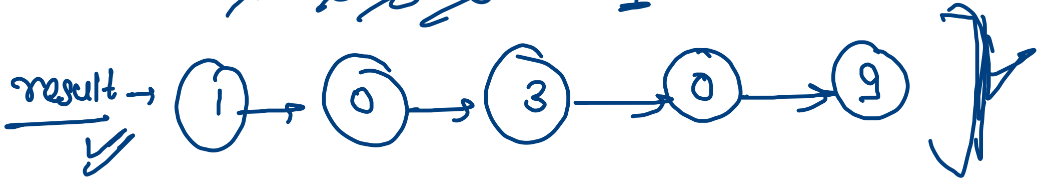
carry = ~~0~~ ~~1~~ ~~1~~ ~~1~~ 0

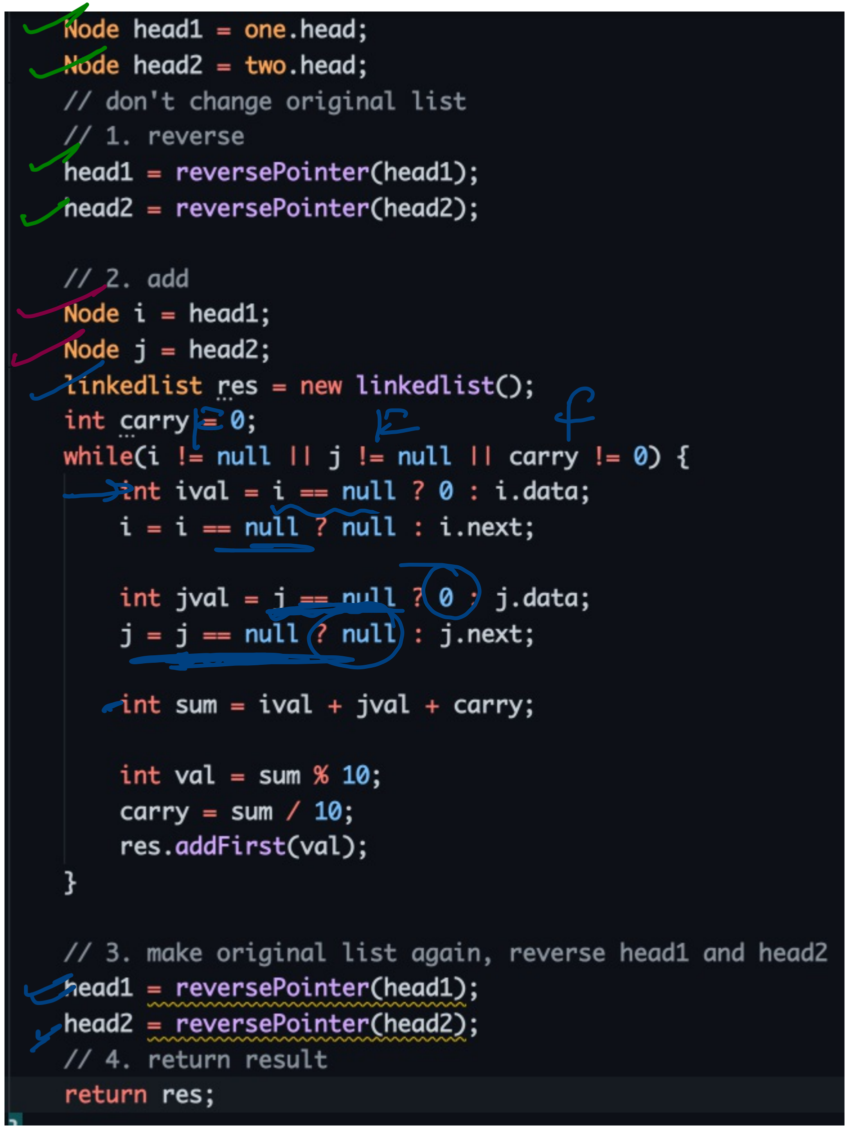
ival = ~~6~~ ~~3~~ ~~4~~ ~~9~~ 0

jval = ~~3~~ ~~7~~ ~~8~~ ~~0~~ 0

sum = ~~9~~ ~~10~~ ~~13~~ ~~10~~ 1

val = ~~9~~ ~~0~~ ~~3~~ ~~0~~ 1

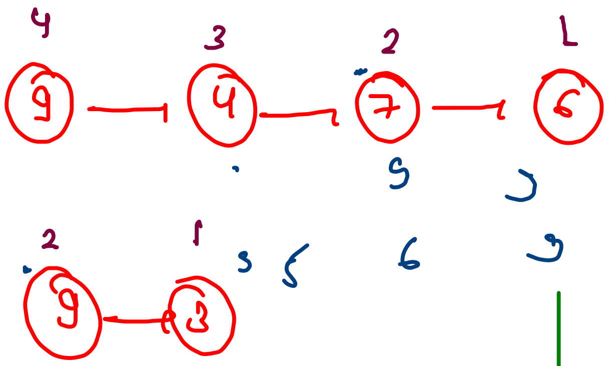




```
Node head1 = one.head;
Node head2 = two.head;
// don't change original list
// 1. reverse
head1 = reversePointer(head1);
head2 = reversePointer(head2);

// 2. add
Node i = head1;
Node j = head2;
linkedlist res = new linkedlist();
int carry = 0;
while(i != null || j != null || carry != 0) {
    int ival = i == null ? 0 : i.data;
    i = i == null ? null : i.next;

    int jval = j == null ? 0 : j.data;
    j = j == null ? null : j.next;

    int sum = ival + jval + carry;

    int val = sum % 10;
    carry = sum / 10;
    res.addFirst(val);
}

// 3. make original list again, reverse head1 and head2
head1 = reversePointer(head1);
head2 = reversePointer(head2);
// 4. return result
return res;
```

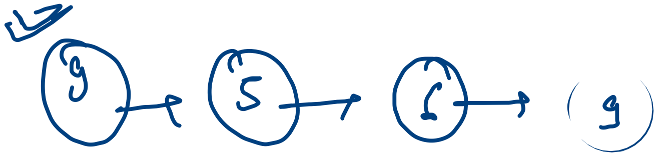
list1 :

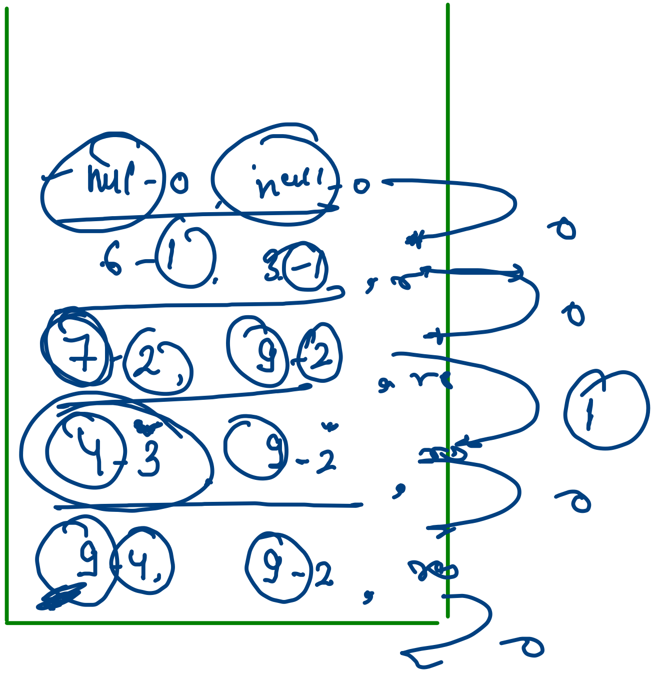
list2 :



return type -> carry :

res = new LinkedList<>()





main

i1 == i2

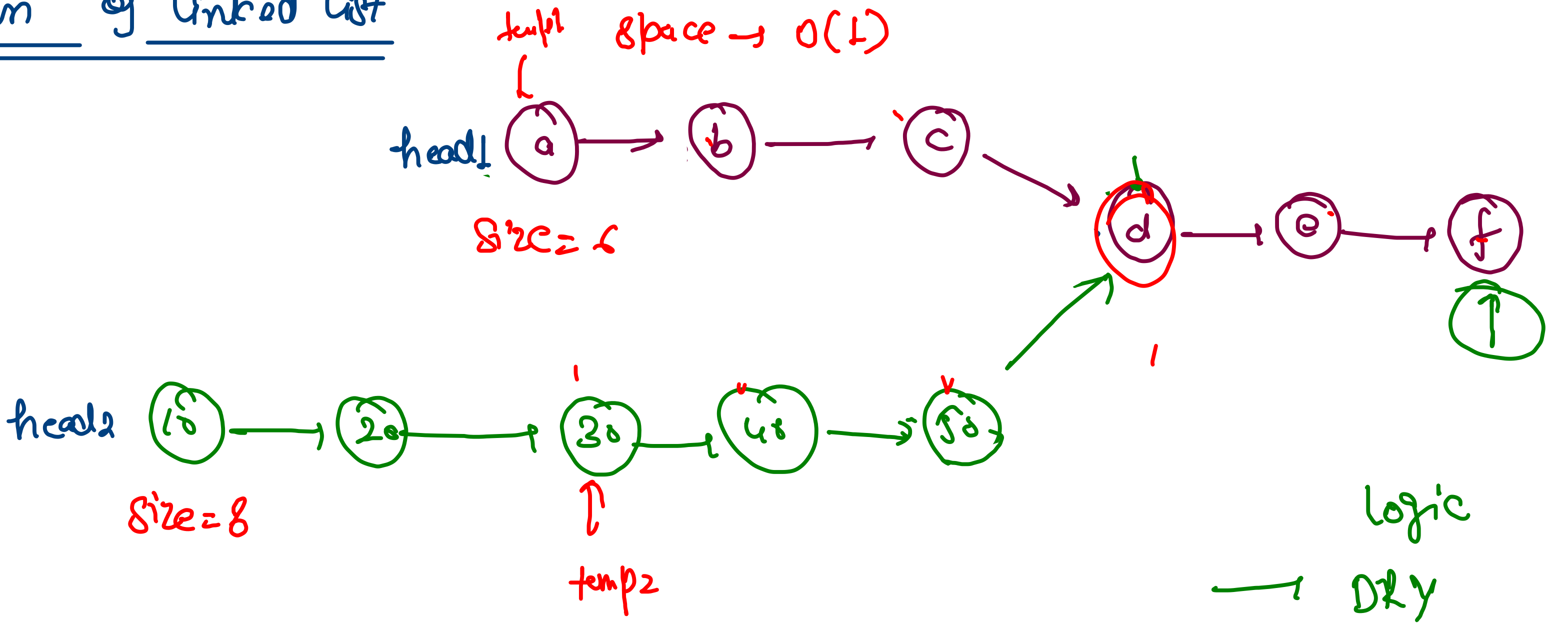
i1 > i2

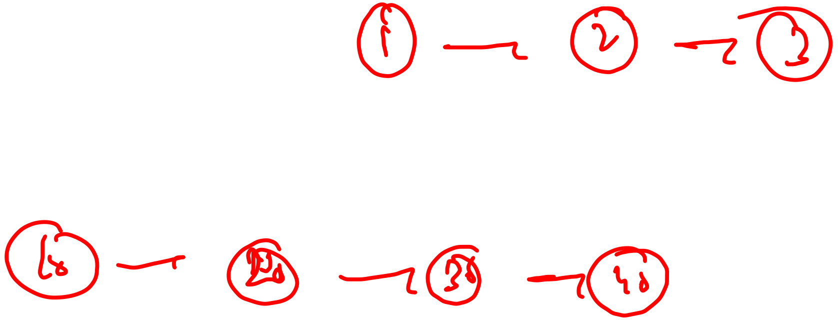
i2 > i1

Intersection of Linked List

time → O(n) × 3 =

space → O(1)





logic

→ DRY

→ Null

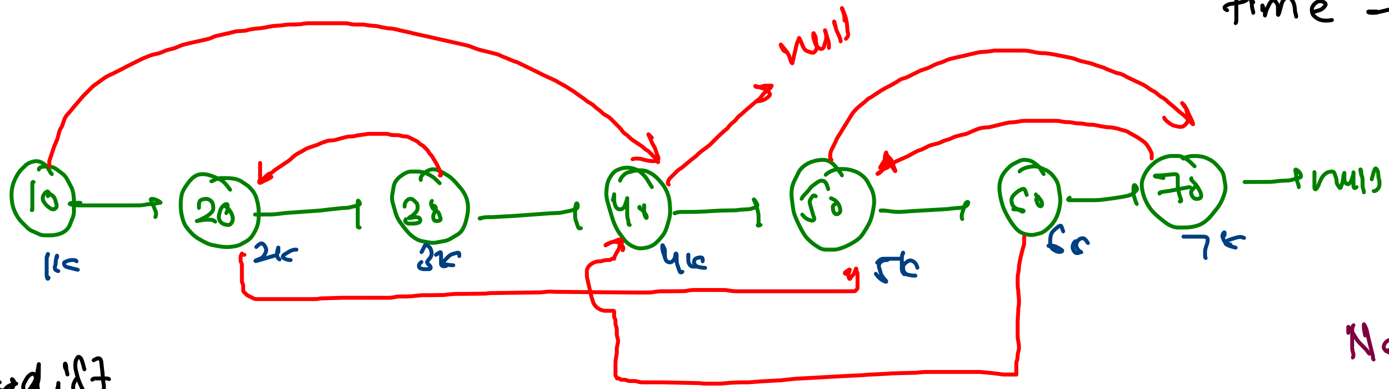
Copy Linked List with Random pointer

Space - O(1) Constant Space

time - O(n)

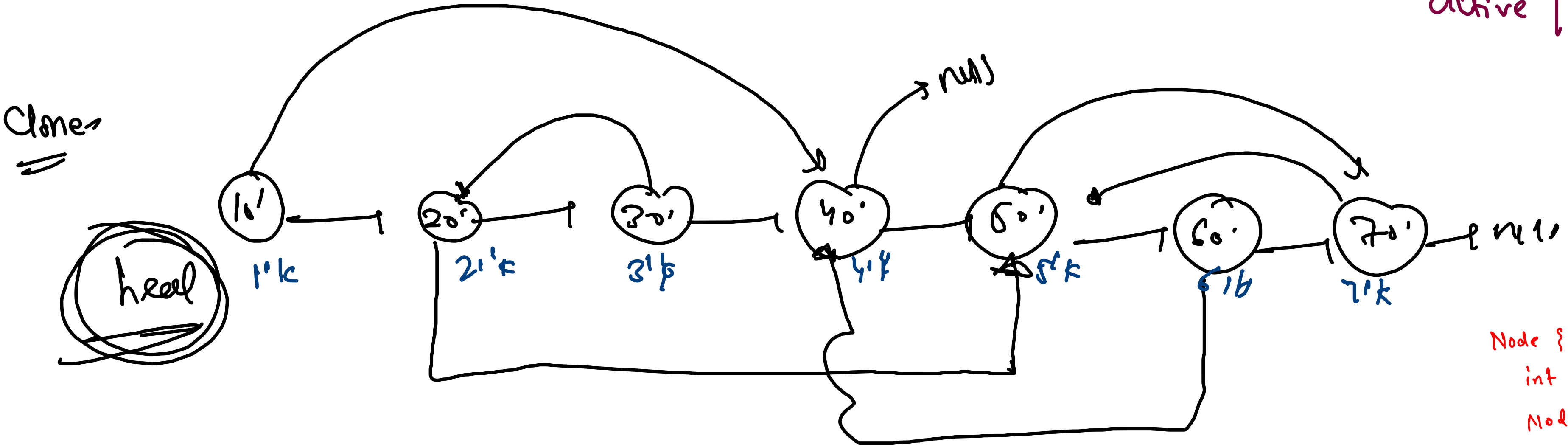
✓

Do not change actual linkedlist



Neurony- active ||✓

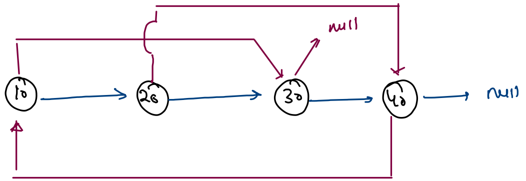
Clones

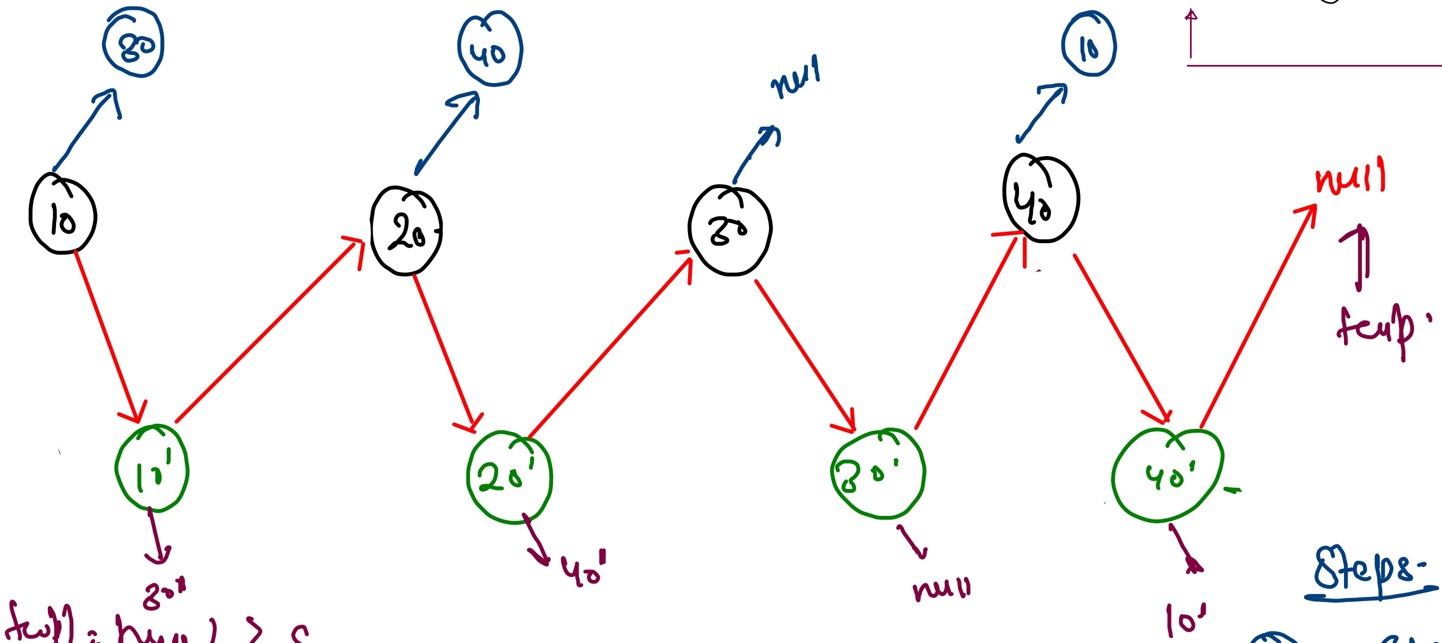


```
Node {
  int data;
  Node next;
  Node random;
}
```

Even + even = Even.

Odd + odd = Even

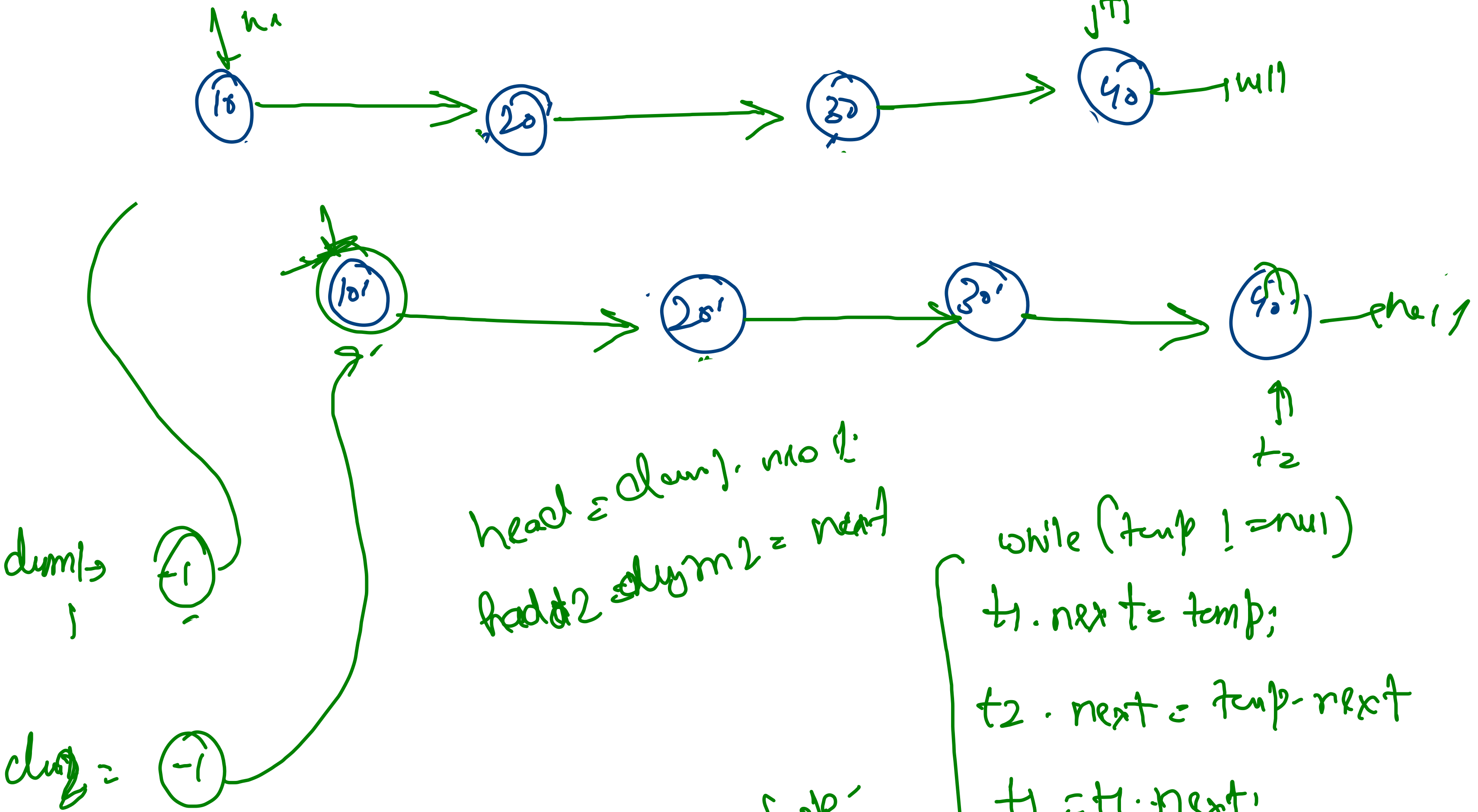




```
while(temp != null) {
    temp.next.random = temp.random == null ? null : temp.random.next;
    temp = temp.next.next;
}
```

Steps-

1. Clone linked with out setup of random pointers
2. Connect in zig-zig
3. Set random pointers, rearrange original.



head = dum1. next;

head2 dum2 = next

```
while (temp != null)
{
    t1.next = temp;
    t2 . next = temp.next
    t1 = t1.next;
    t2 = t2.next.
    temp = temp. next.next
}
```

loop

t1.next = null
t2.next = null