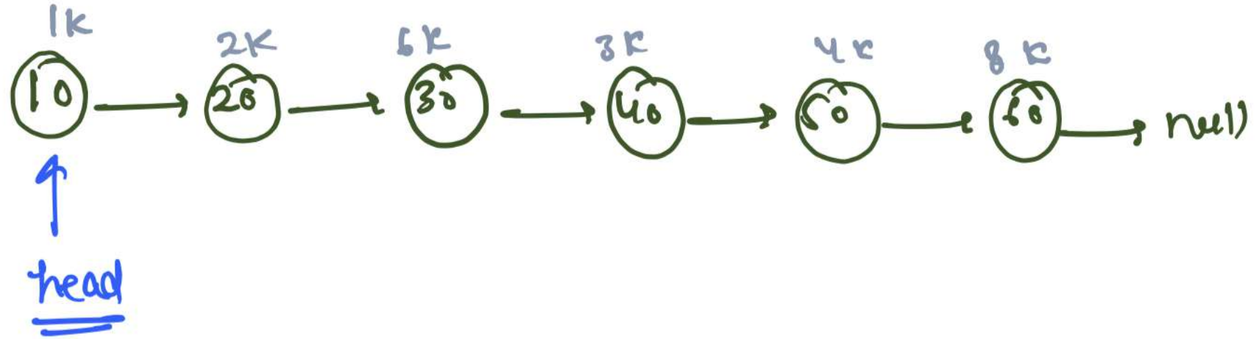
Reverse Linked List

Tuesday, 3 August 2021 5:55 PM

Leetcode 206. https://leetcode.com/problems/reverse-linked-list/

Time complexity → O(n)
space complexity → constant

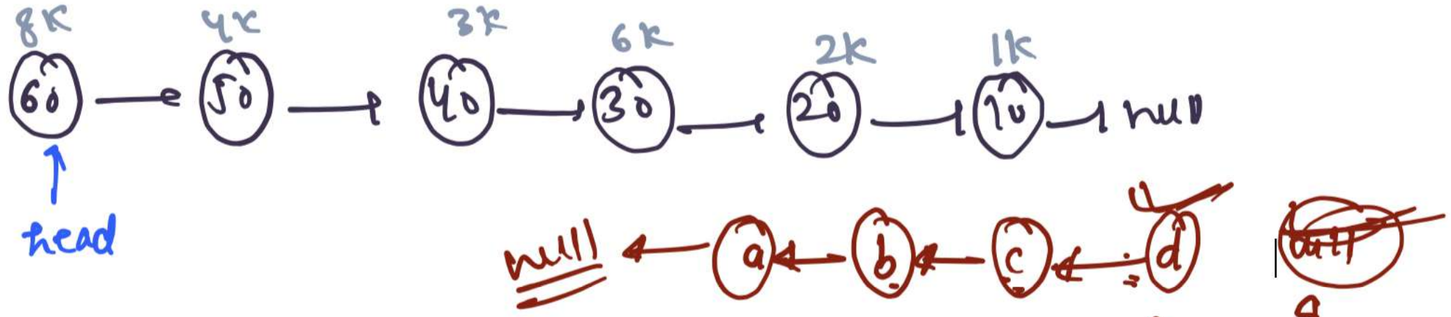
Input



NOTE:→ add of nodes are same in input and output the connection is reverse.

→ pointer Reverse.

output



prev, curr, next

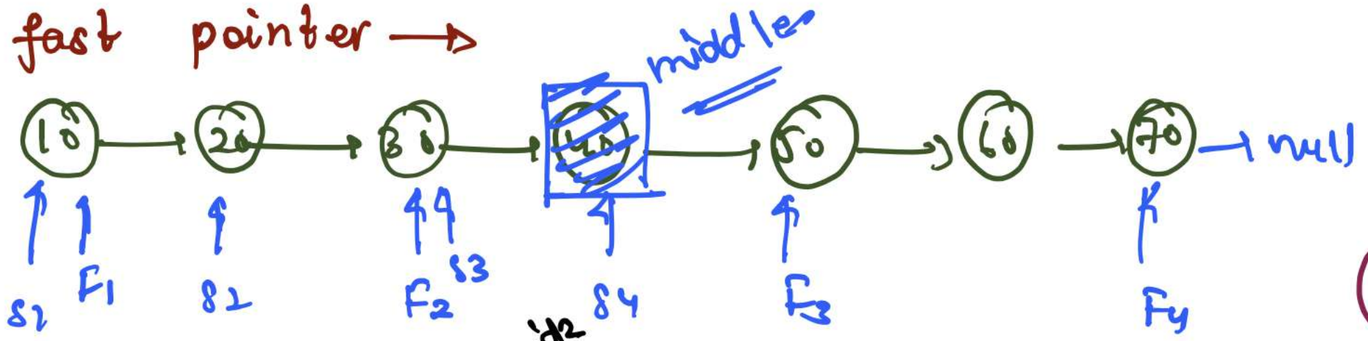
0 next = current.next
1 curr.next = prev.
2 prev = curr
3 curr = next

Middle Of LinkedList

Leetcode 876 https://leetcode.com/problems/middle-of-the-linked-list/

Tuesday, 3 August 2021 5:56 PM

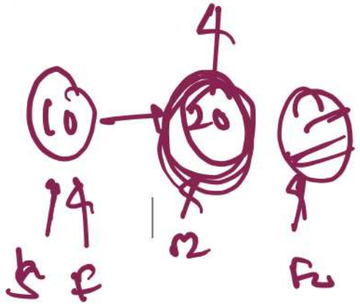
using slow and fast pointer →



odd

slow = head

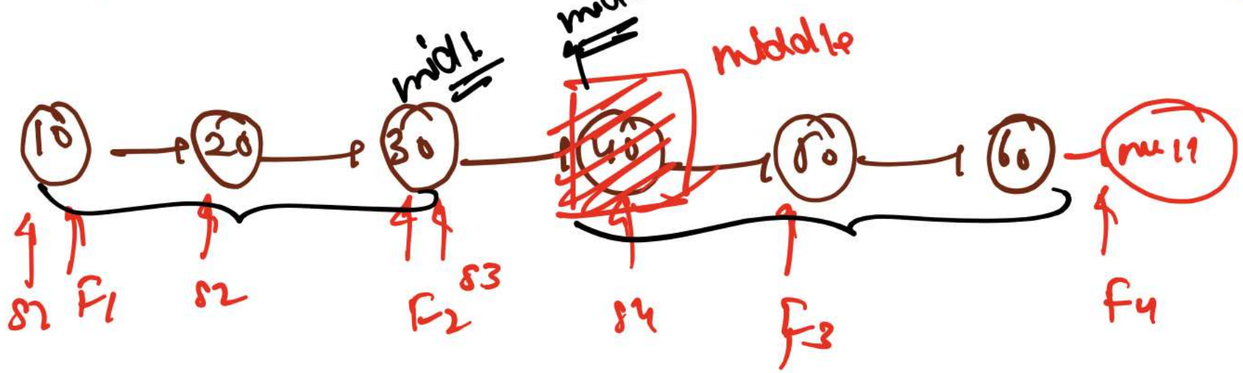
fast = head



Even

slow = head

fast = head



slow = head

fast = head.next

Mid 1

from

slow = head

fast = head

middle = 2

condn of Running

while(fast != null && fast.next != null)

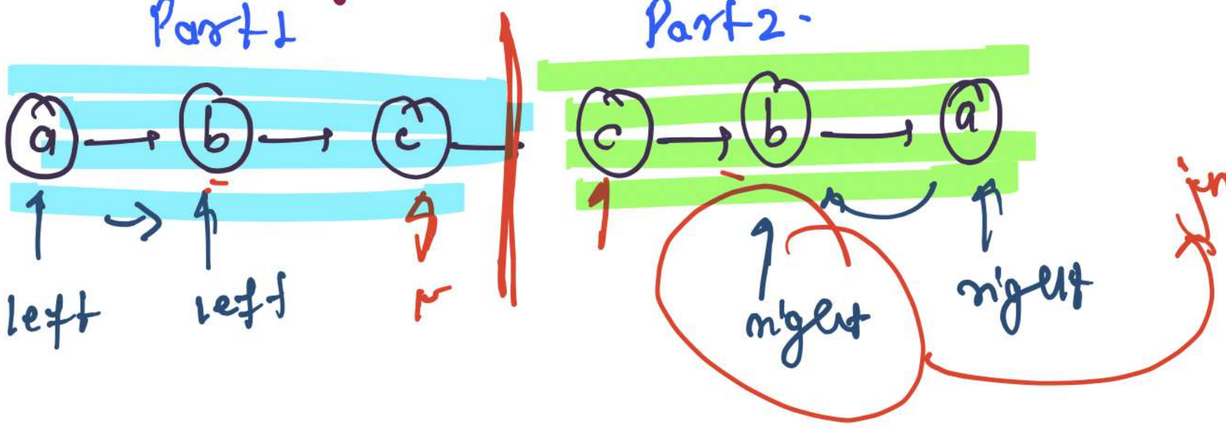
Palindrome LinkedList

Leetcode 234. https://leetcode.com/problems/palindrome-linked-list/

Tuesday, 3 August 2021 5:58 PM

allowed time complexity → O(n)

allowed space complexity → constant



jump in prev point in singly linked list is of complexity → O(n)

→ $\frac{n}{2} \times O(n) = O(n^2)$

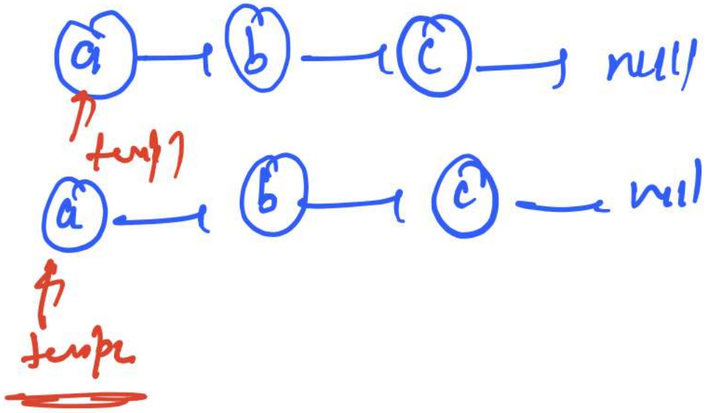
arr[left] == arr[right]

left ++

right --

Part 1 →

(Part 2)' → Reverse

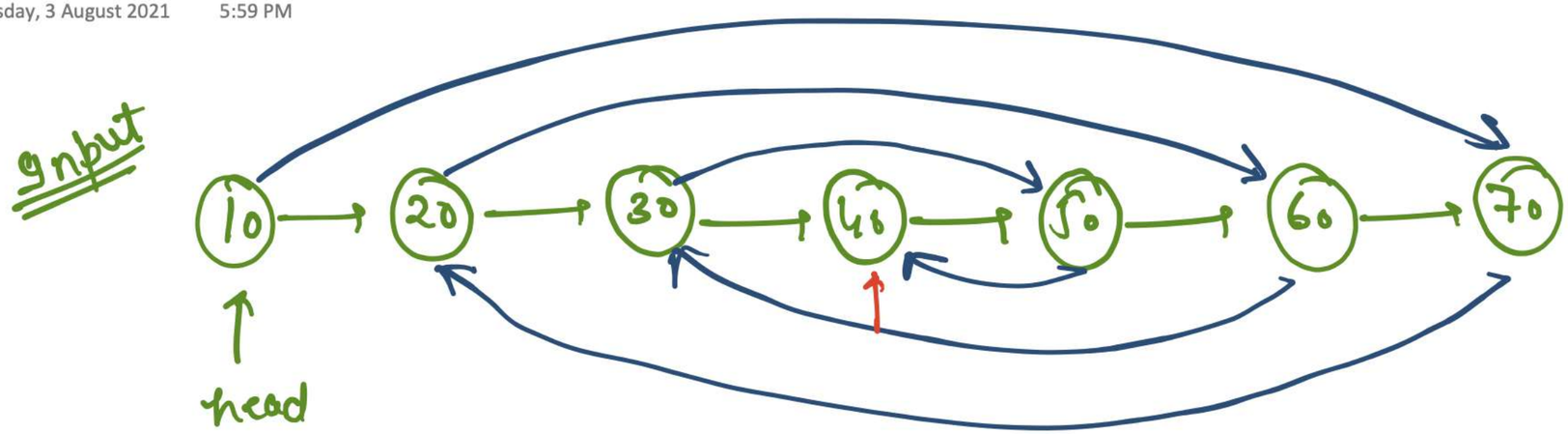


while both are not equal to null (& data is equal)

Fold Of a LinkedList

Leetcode 143. https://leetcode.com/problems/reorder-list/

Tuesday, 3 August 2021 5:59 PM



steps for solution

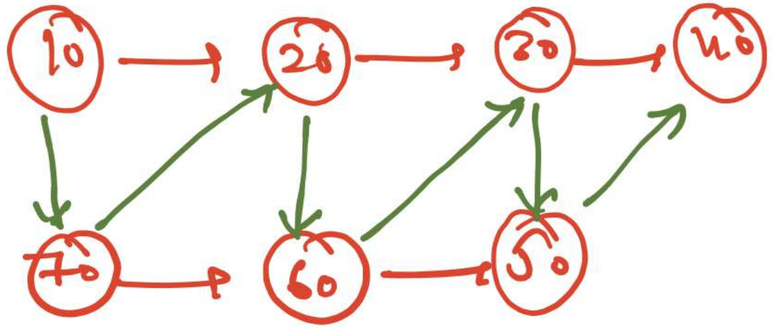
1. Segregate first half and second half
2. Reverse second half
3. Make connection with proper logic to match output.

output → 10 → 70 → 20 → 60 → 30 → 50 → 40 → null

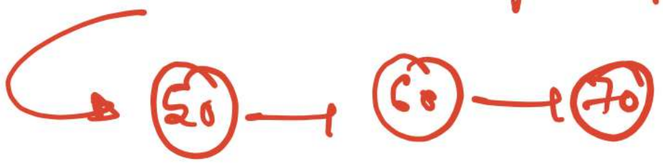
matching

solution →

first half of input →



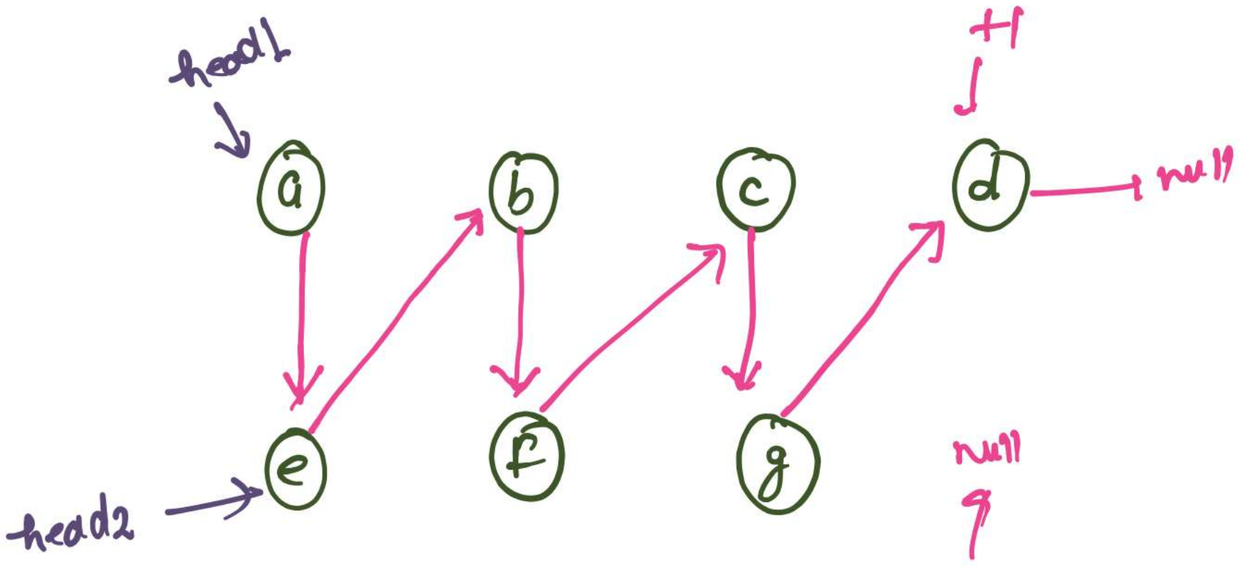
second half



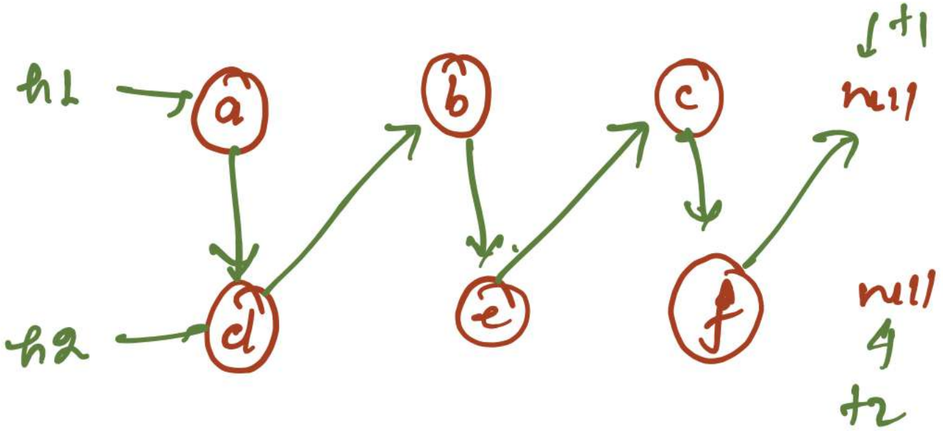
reverse

10 → 70 → 20 → 60 → 30 → 50 → 40 → null

connection Management if size is odd →



connection Management is size is even →



Running check →

```
while( temp1 != null && temp2 != null)
{

next1 = temp1.next;
next2 = temp2.next;
temp1.next = temp2
temp2.next = next1
temp1 = next1;
temp2 = next2;

}
```

Unfold A LinkedList

From Portal : Pepcoding

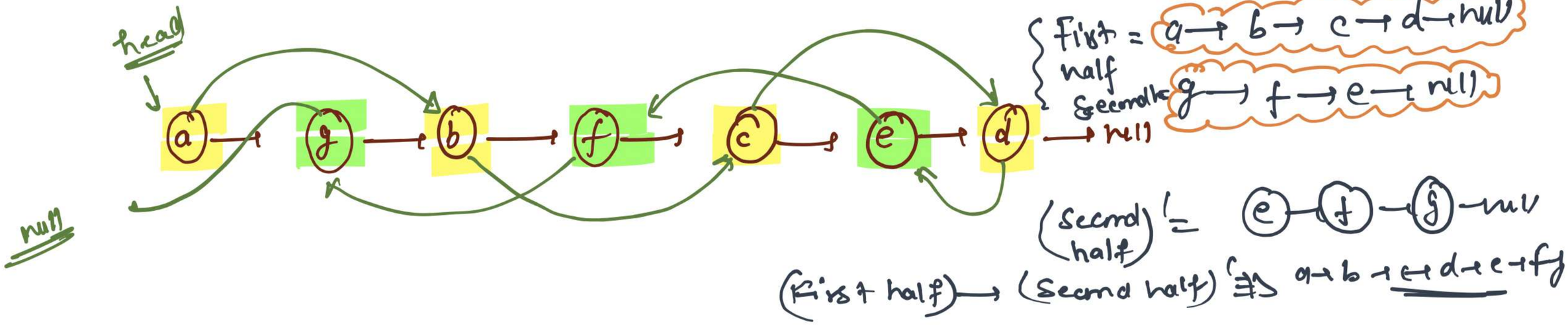
Tuesday, 3 August 2021 6:00 PM

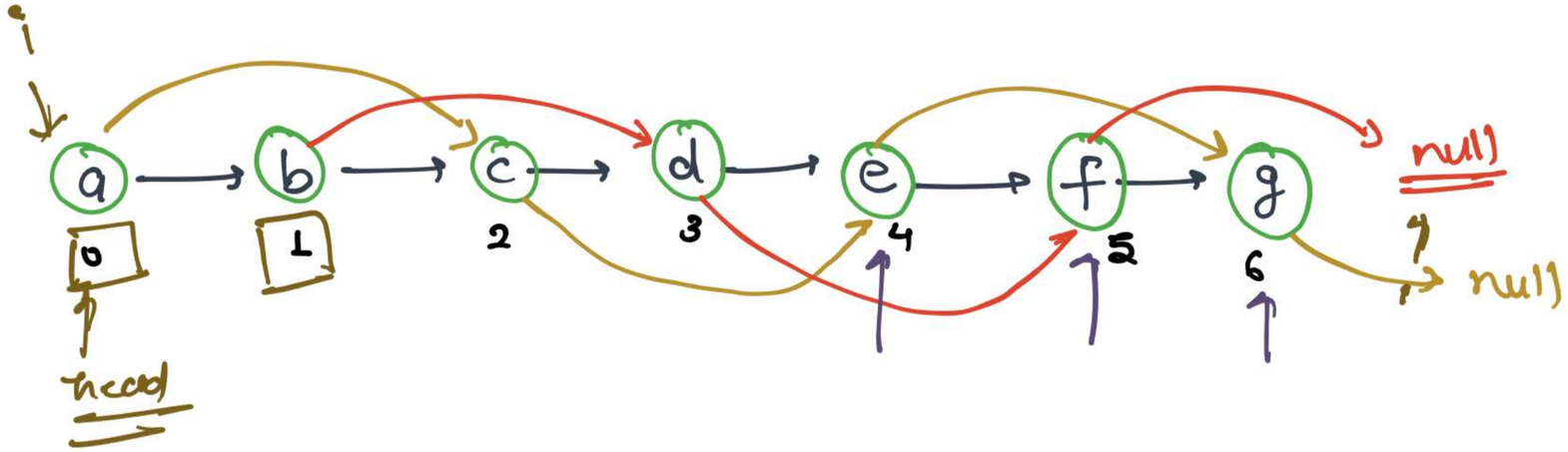
Normal linked list → a → b → c → d → e → f → g → null

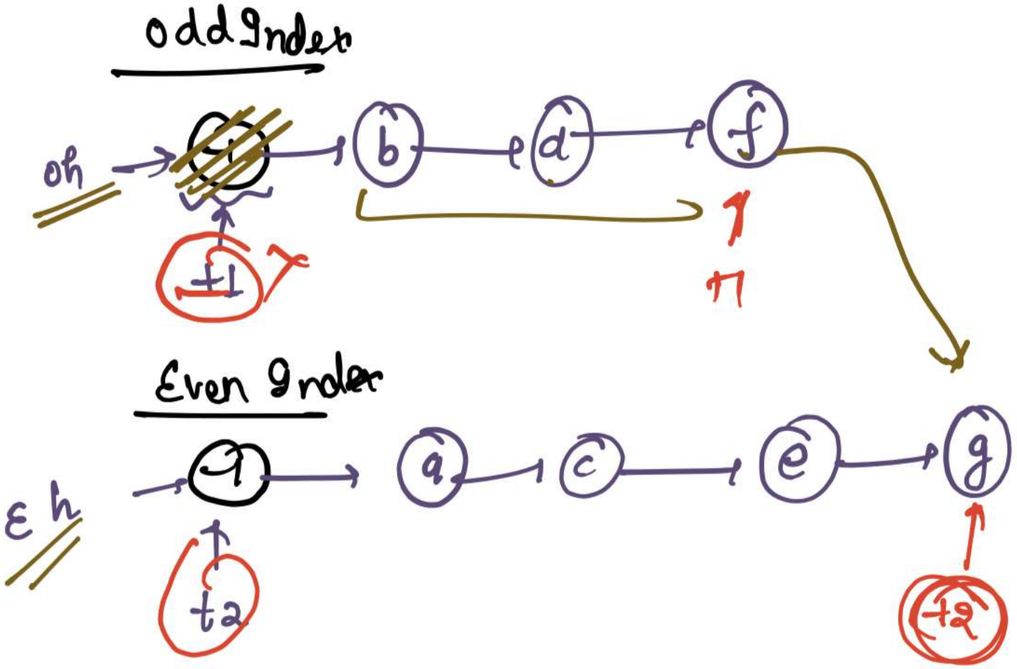
input → Fold → a → g → b → f → c → e → d → null

output → Normal form of fold

i.e. a → b → c → d → e → f → g → null







Reverse ⇒ ohead = reverse(oh.next)

next to avoid dummy head

oh: f → d → b → null

t2.next = ohead.

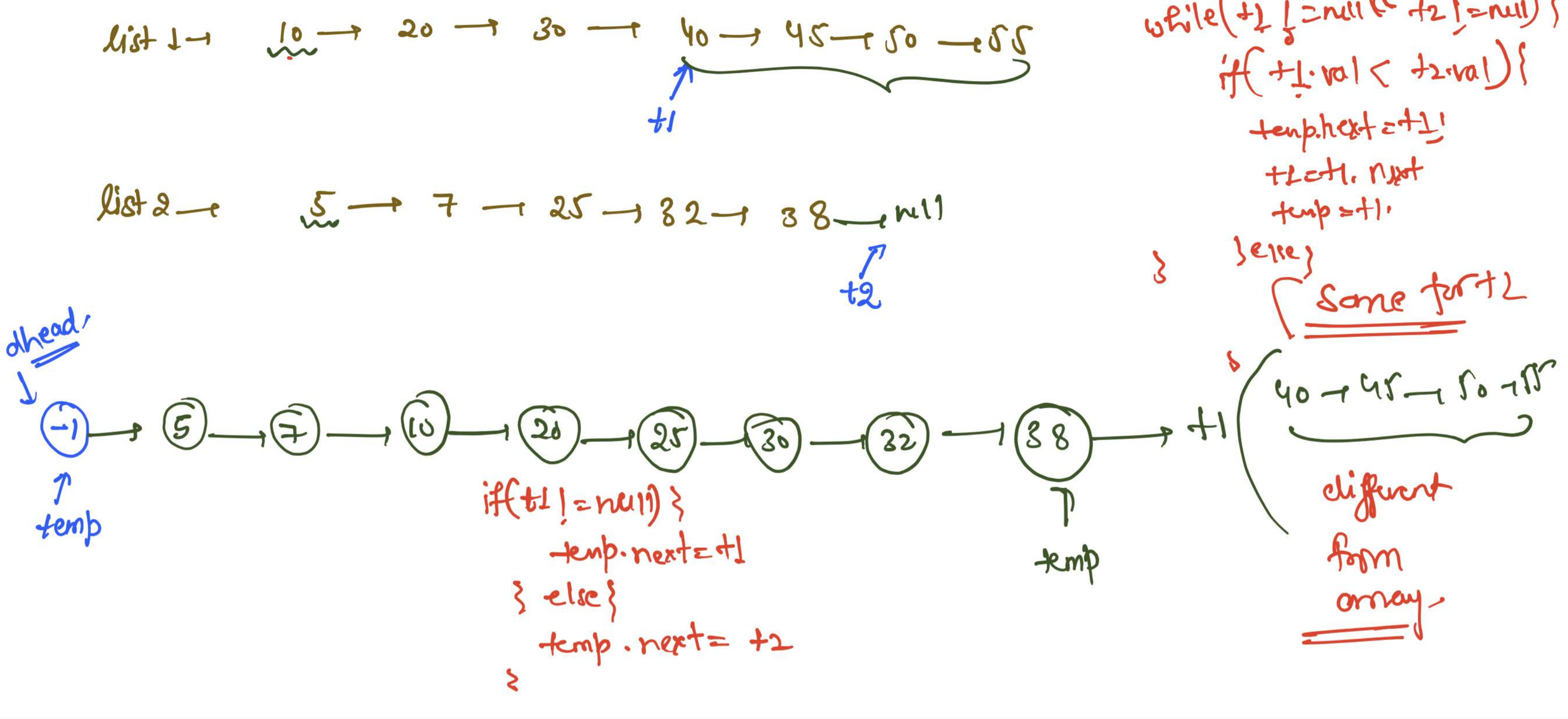
Steps:

1. Make two dummy Nodes.
2. Iterate and make connection according to odd & Even Index

Merge Two Sorted LinkedList

Tuesday, 3 August 2021 6:02 PM

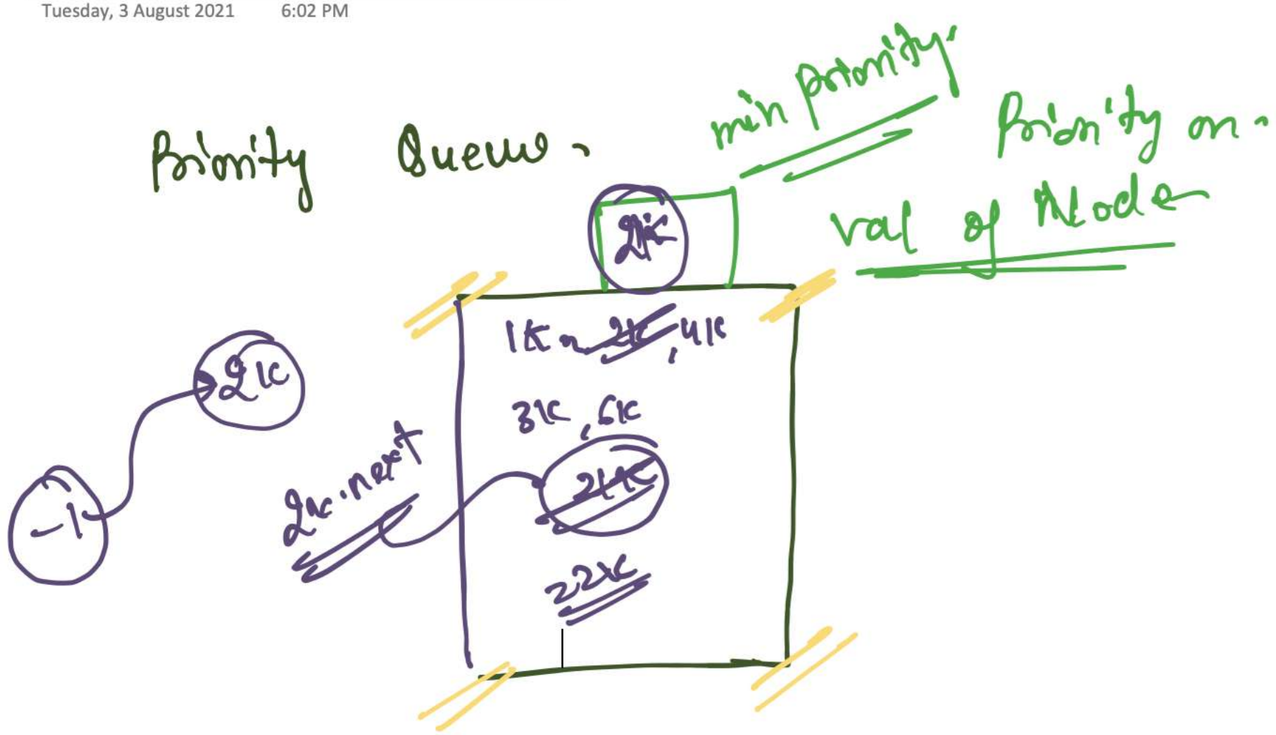
Leetcode 21. https://leetcode.com/problems/merge-two-sorted-lists/

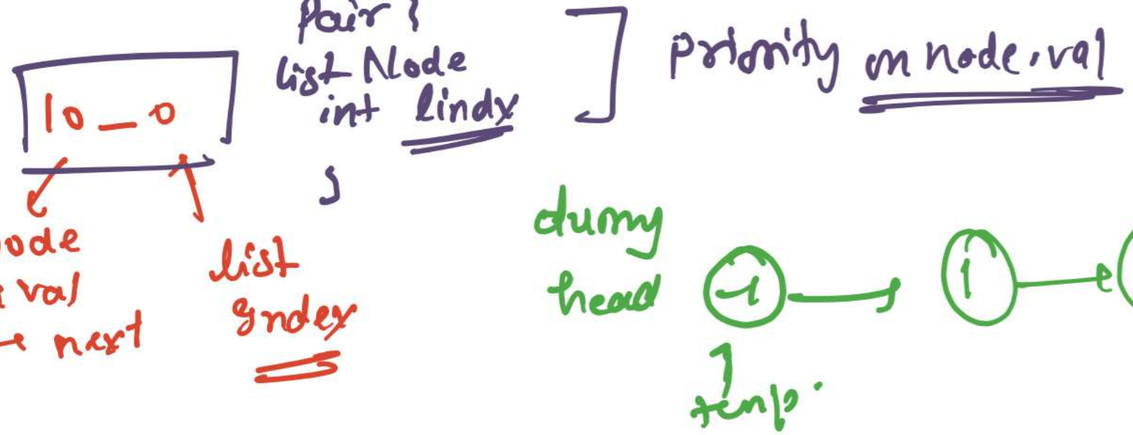


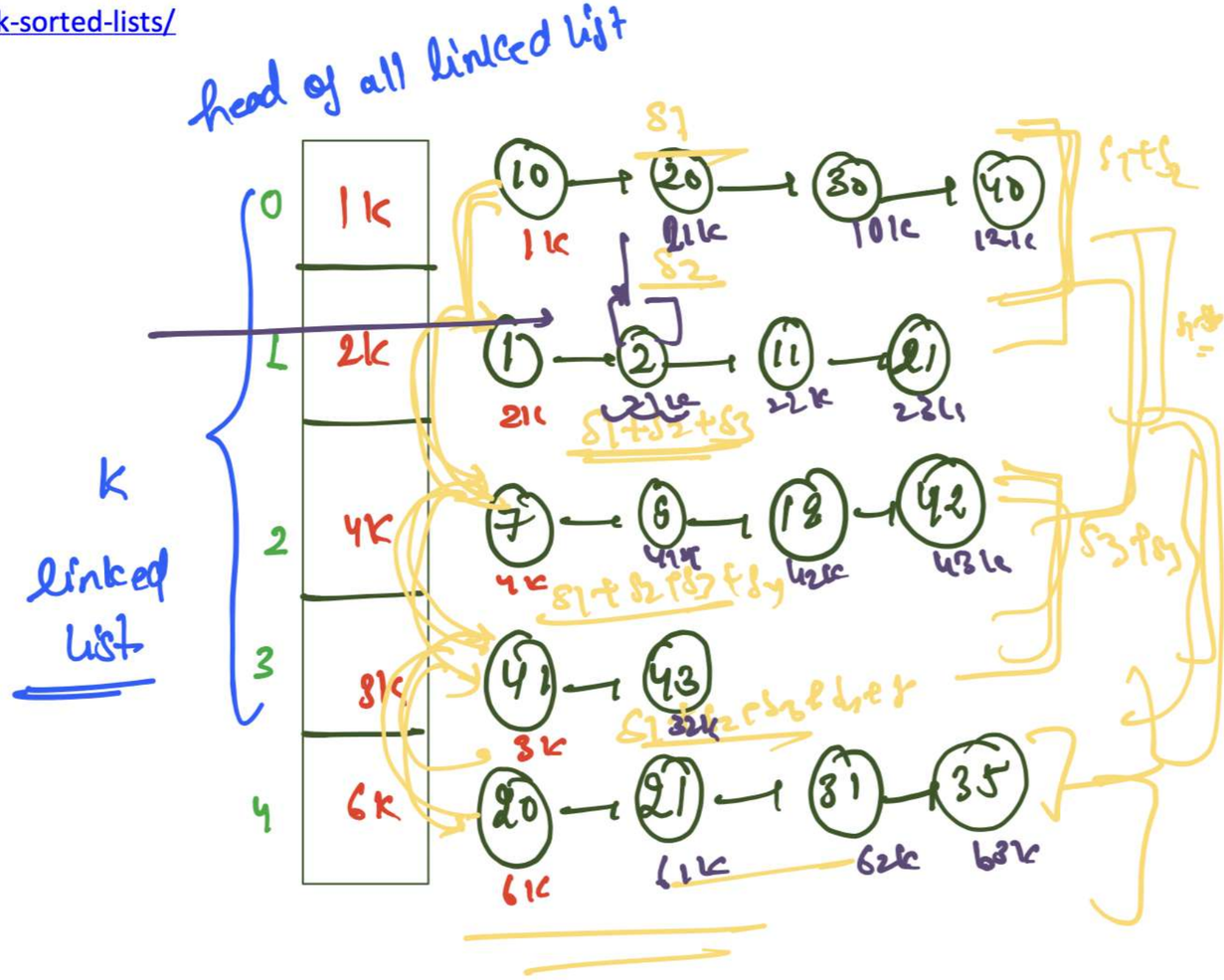
Merge K Sorted LinkedList

Tuesday, 3 August 2021 6:02 PM

Leetcode 23. https://leetcode.com/problems/merge-k-sorted-lists/







Merge Sort LinkedList

Tuesday, 3 August 2021 6:04 PM

Leetcode 148. https://leetcode.com/problems/sort-list/