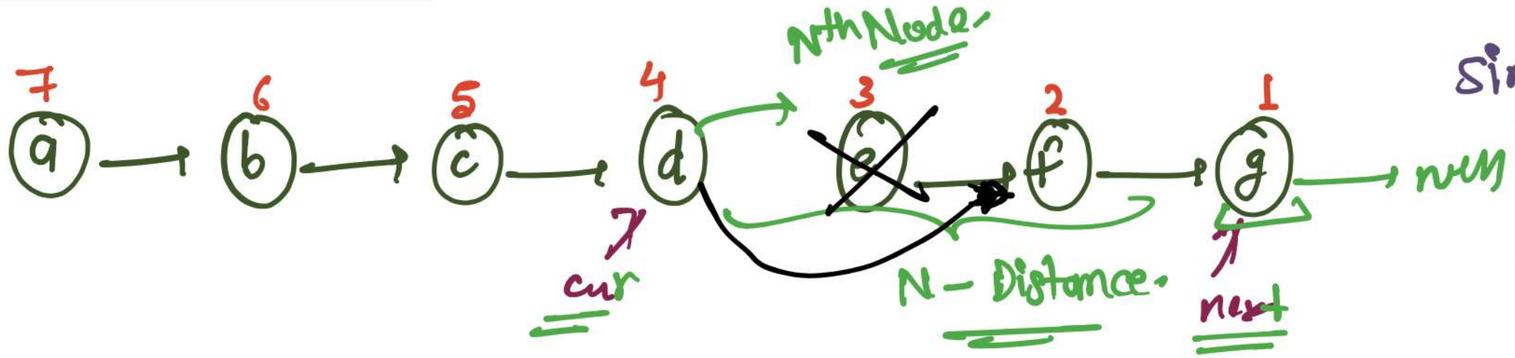
Remove Nth Node from End of LinkedList

Leetcode 19 https://leetcode.com/problems/remove-nth-node-from-end-of-list/

Thursday, 5 August 2021 8:08 PM



null
↑
curr

Single Iteration

Remove third last

n=3

Step →
1. N steps move by next pointer
2. curr point to head
3. Move curr and next simultaneously

next.next == null

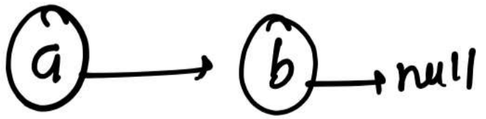
N == size

return head.next

next.next → End of linked list

curr.next → Nth Node

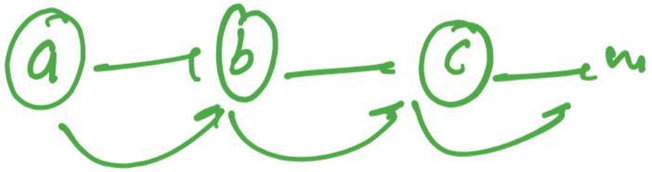
curr.next = curr.next.next



next pointer.
curr. pointer.

1. Null
2. head.next == null → Single Node.
3. head.next.next == null → Double Node.
4. Odd Size check
5. Even Size check

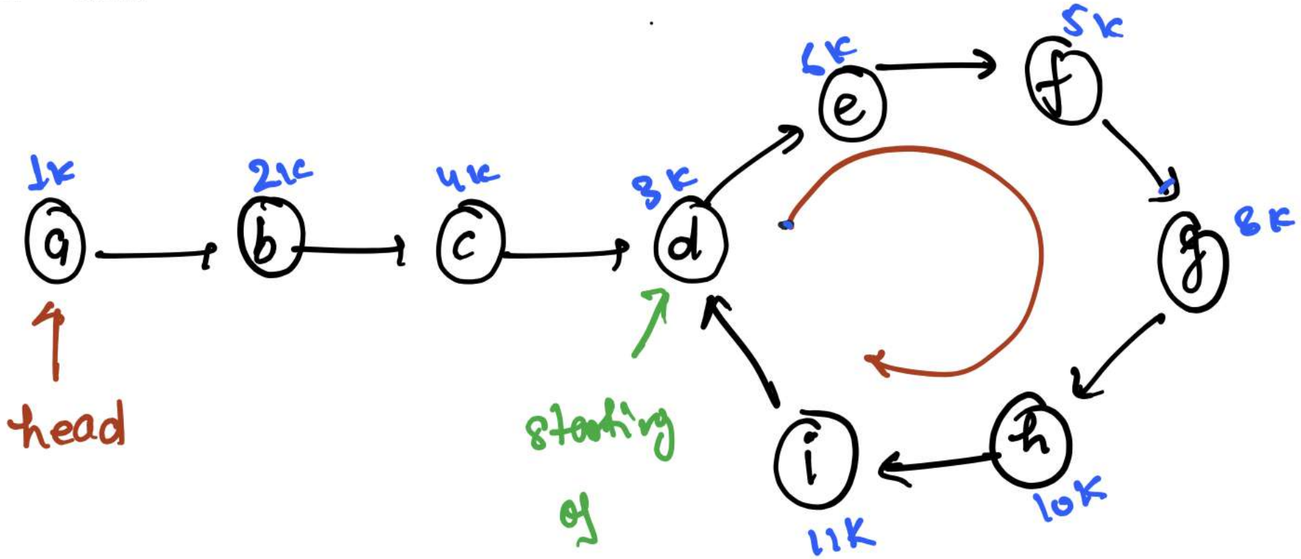
→ Sometime this is needy check



Is Cycle Present in the LinkedList

Thursday, 5 August 2021 8:09 PM

Leetcode 141 https://leetcode.com/problems/linked-list-cycle/



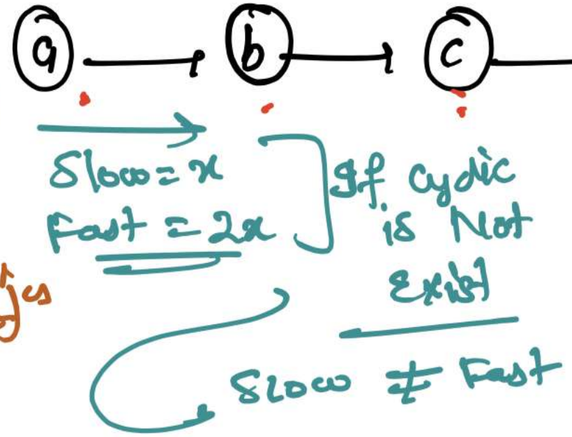
Floyd Cycle Detection Method

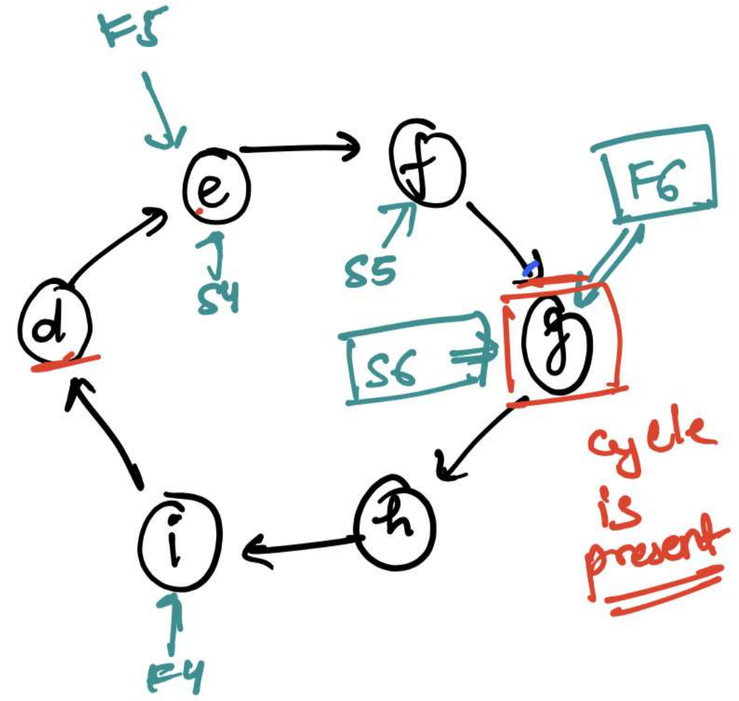
Mathematics is language of physics.

Slow = head ⇒ speed = x

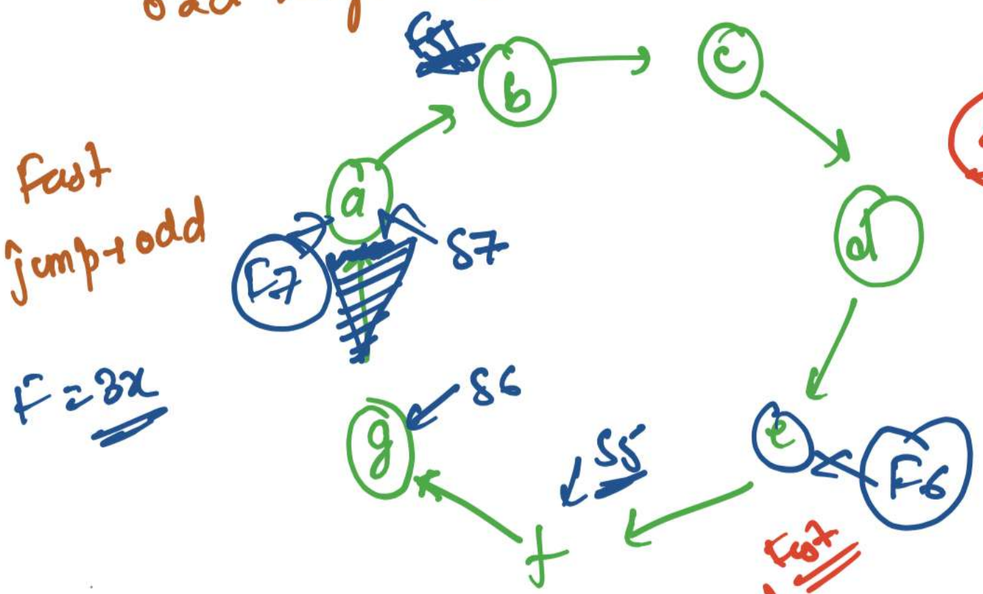
Fast = head ⇒ speed = 2x → Even

3x, 4x → Evenjes





odd length cycle



Fast jump → odd

F = 3x

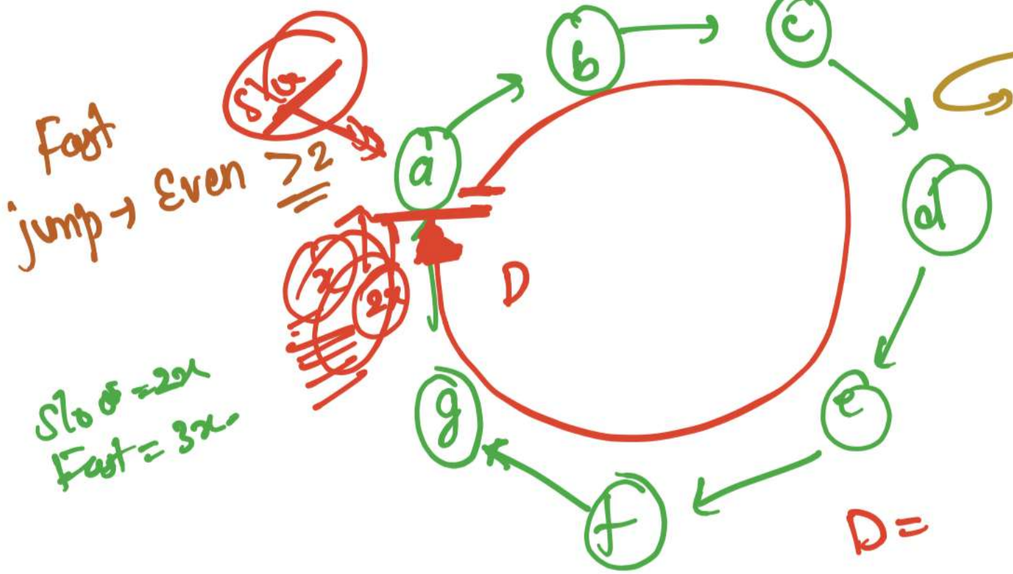
slow × Fast-d

optimal is

Slow = x

Fast = 2x

Beacuse in this case slow take single Rotation =



Fast jump → Even ≥ 2

Slow = 2x

Fast = 3x

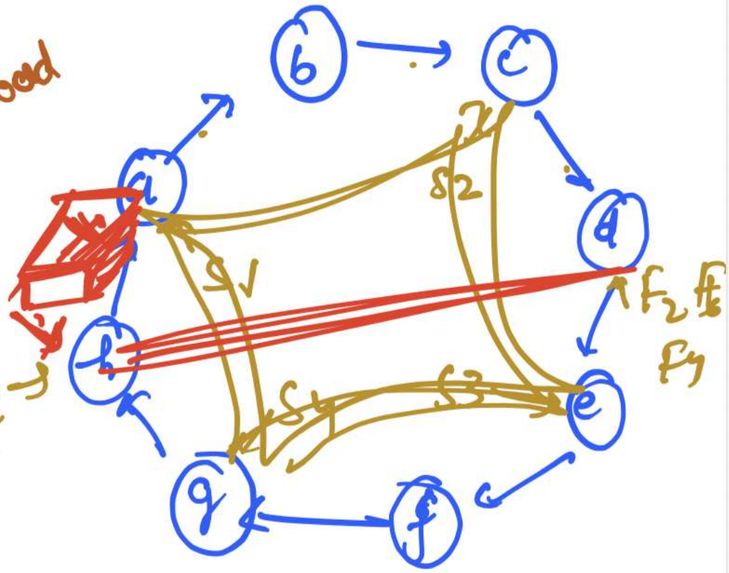
D =

Even length cycle.

Fast jump → odd

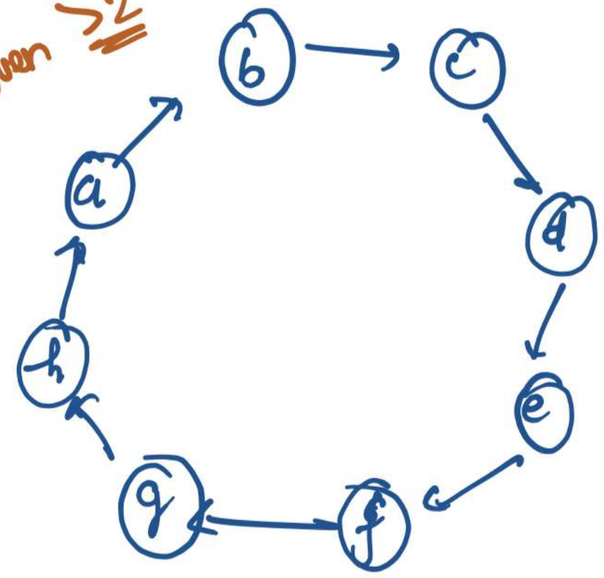
S = 2x

F = 4x



Even length cycle

Fast jump → Even ≥ 2

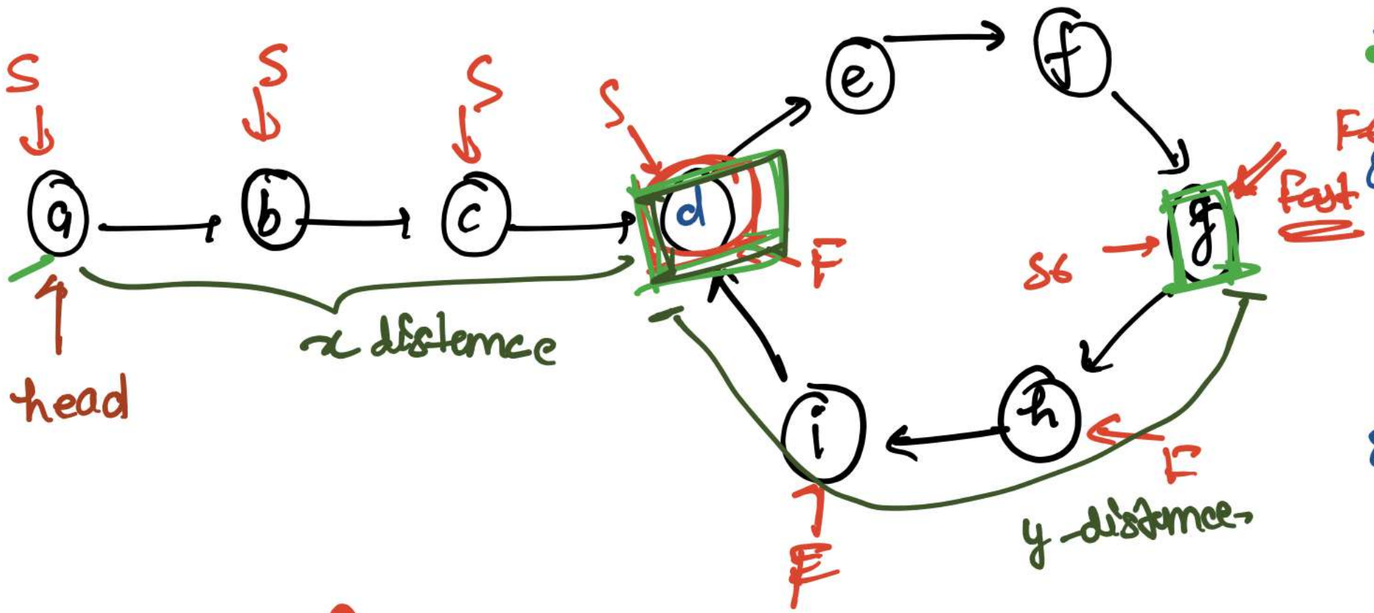


jump = Even.

Cyclic Node : Starting of Cycle

Leetcode 142 https://leetcode.com/problems/linked-list-cycle-ii/

Thursday, 5 August 2021 8:09 PM



Step 1 → Slow = head
Fast = head

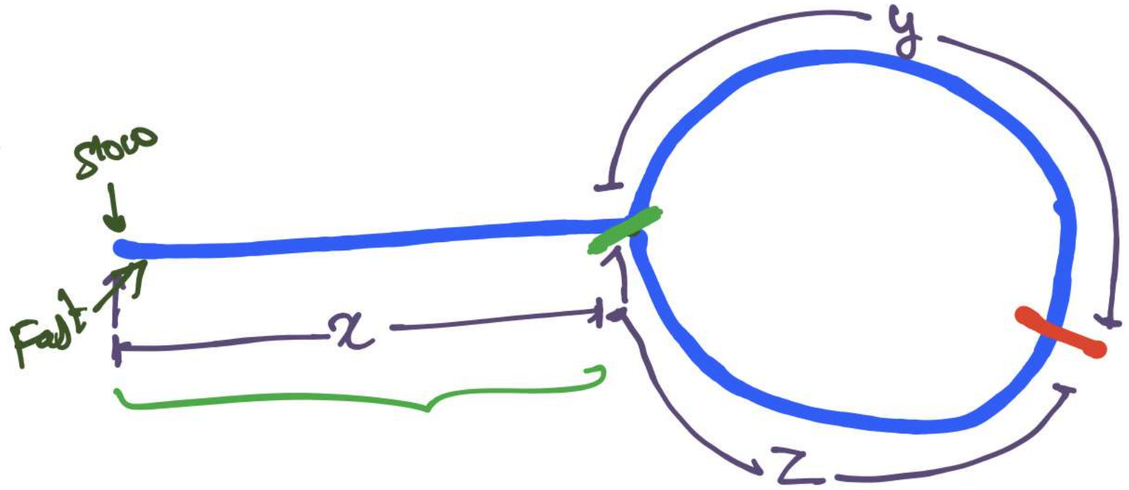
Step 2 → Move slow with 'x'
Move fast with '2x'
until they don't Meet

Step 3 → Slow = head, Fast = meeting point
→ move slow and fast with x speed until they don't meet

Step 4 → Meeting point is answer

(x = y) distance

What of problem



To prove ⇒ $x = z$ (cyclic linked list)

Speed of slow = s

Speed of Fast = $2s$

time taken by slow and Fast = T

distance travel by slow = $x+y$

distance travel by Fast = $x + r(y+z) + y$

r - rotation of cycle.

$$\text{Speed} = \frac{\text{distance}}{\text{time}}$$

slow → $s = \frac{x+y}{T}$ — (1)

Fast ⇒ $2s = \frac{x + r(y+z) + y}{T}$

$$s = \frac{x + r(y+z) + y}{2T} \quad \text{— (2)}$$

$$\delta = \frac{x+y\cdot}{T} \quad \text{---- (1)} \qquad \delta = \frac{x+r(y+z)+y\cdot}{2T} \quad \text{---- (2)}$$

From eq (1) and (2)

$$\Rightarrow \frac{x+y\cdot}{T} = \frac{x+r(y+z)+y\cdot}{2T}$$

$$\Rightarrow 2x+2y = x+r(y+z)+y\cdot$$

$$x = r(y+z) - y\cdot \quad , \quad \text{Add } z \text{ and remove}$$

$$x = r(y+z) - y - z + z$$

$$x = (r-1)(y+z) + z$$

Displacement = 0 for any Rotation ≥ 1

for $r=1$

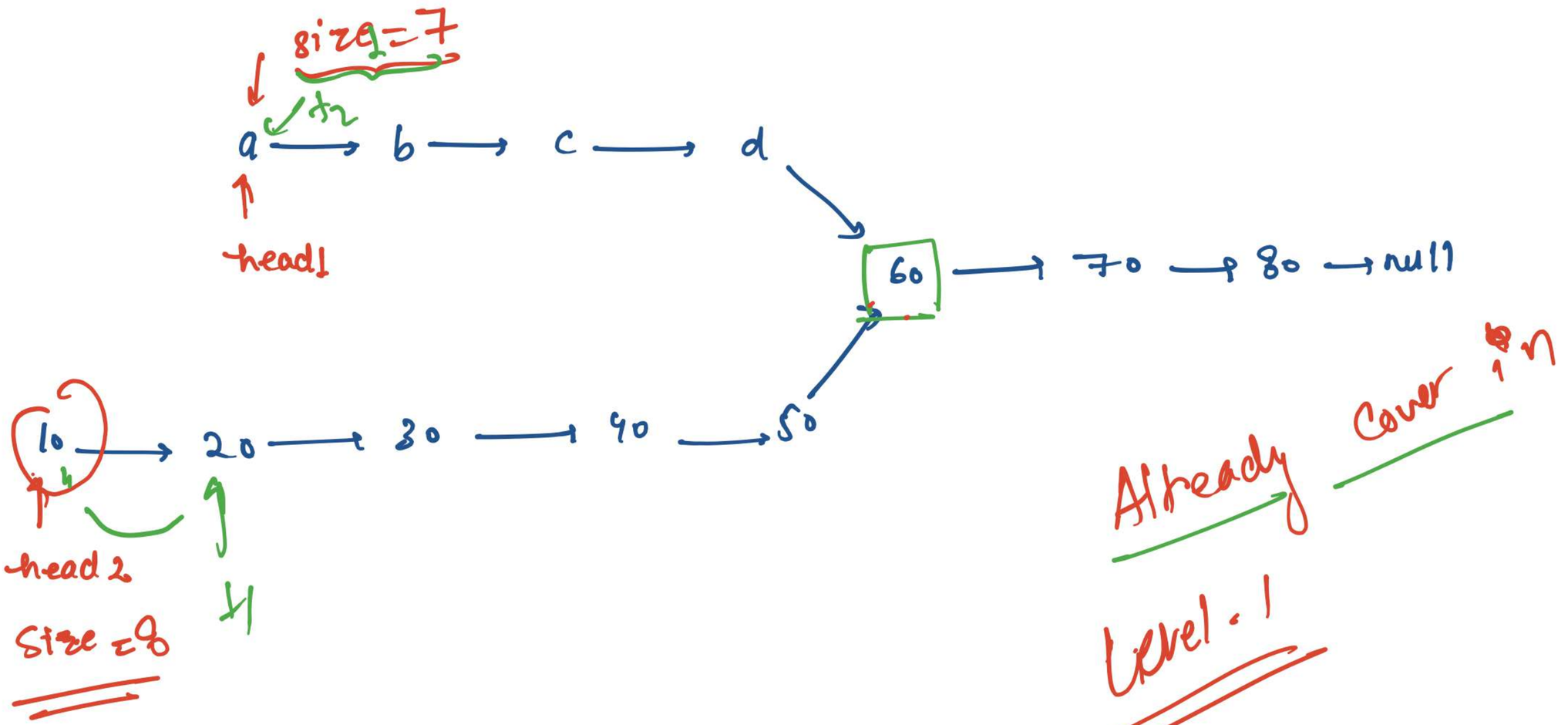
$$x = 0 + z$$

$$\Rightarrow \boxed{x = z} \text{ proved}$$

Intersection Point Of LinkedList : Difference Method

Leetcode 160 https://leetcode.com/problems/intersection-of-two-linked-lists/

Thursday, 5 August 2021 8:09 PM



Intersection Point Of LinkedList : Cycle Method

Leetcode 160 https://leetcode.com/problems/intersection-of-two-linked-lists/

Thursday, 5 August 2021 8:09 PM