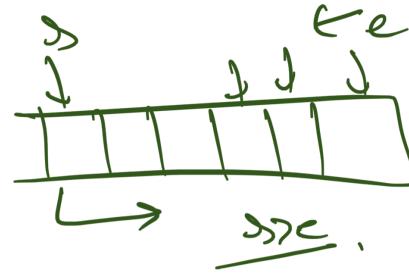
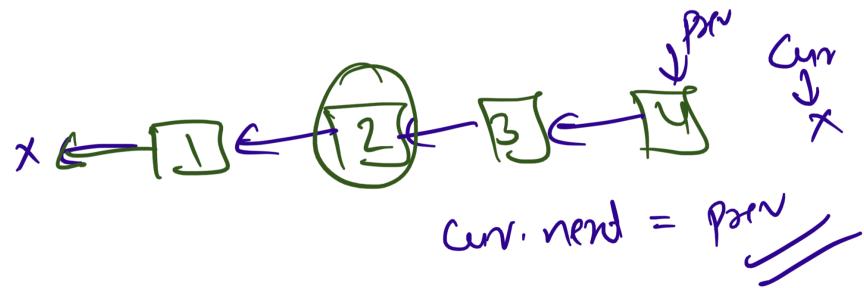
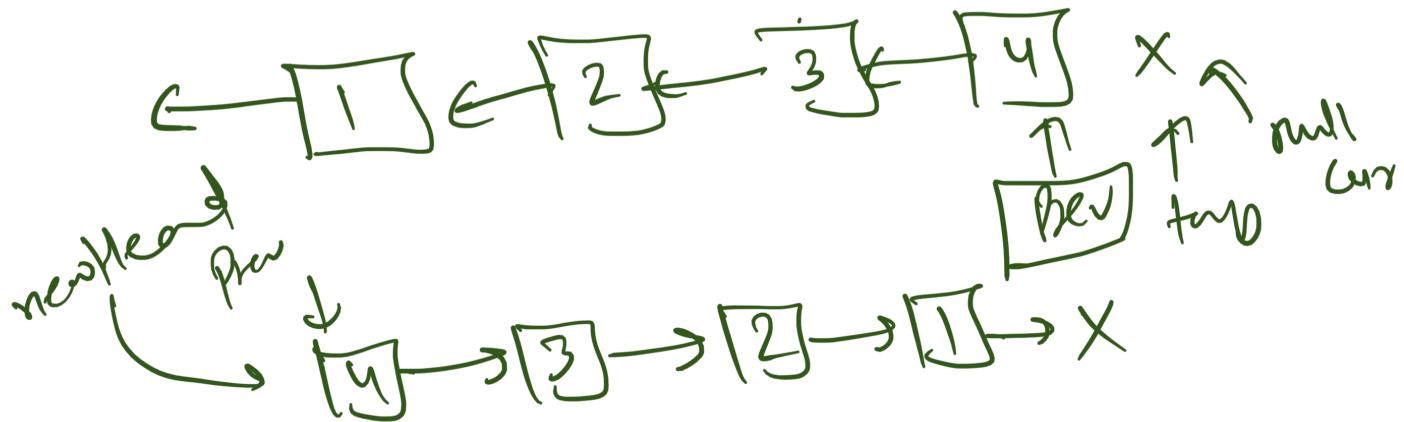
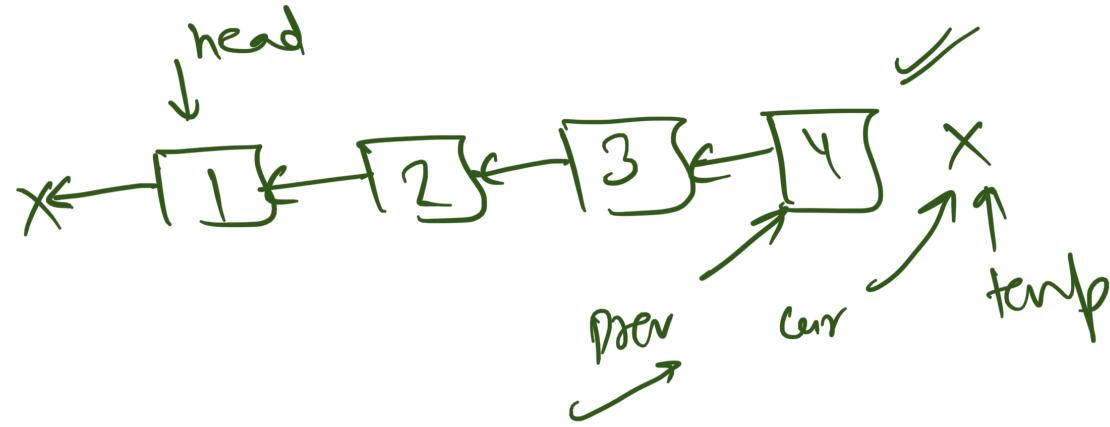


Linked List Problems - I

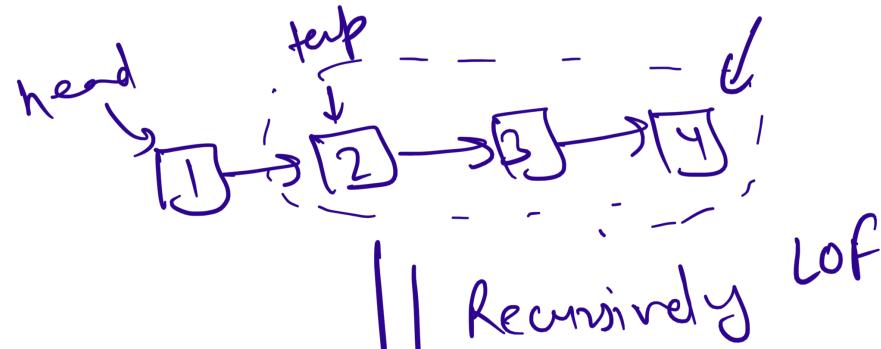
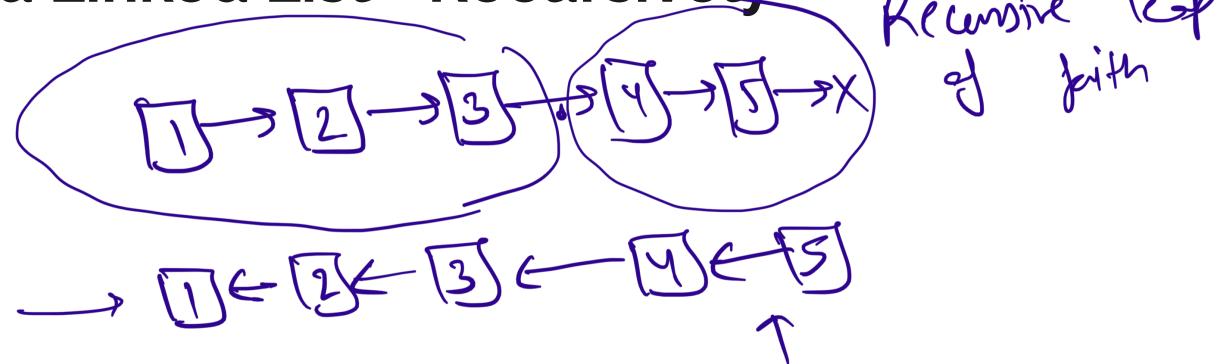


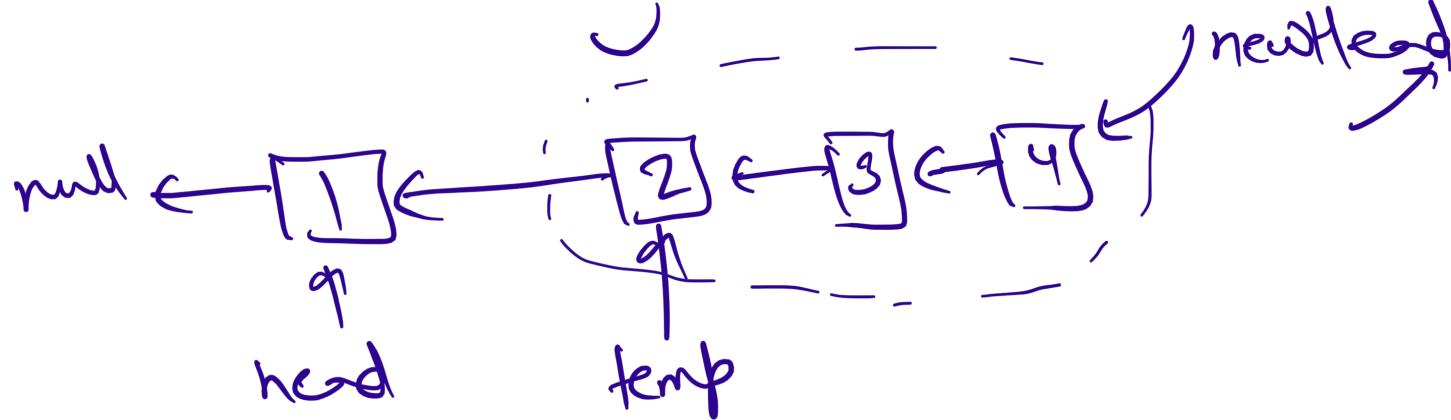
Reverse a Linked List - Iteratively





Reverse a Linked List - Recursively



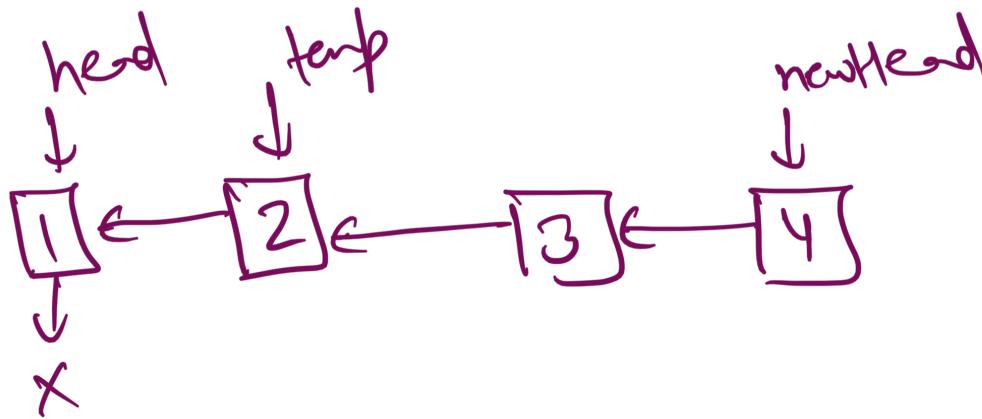


\uparrow^4
 $\rightarrow \text{genode}(1)$

\downarrow^{\uparrow^4}
 $\hookleftarrow \text{genode}(2)$

\downarrow^{\uparrow^4}
 $\text{genode}(3)$

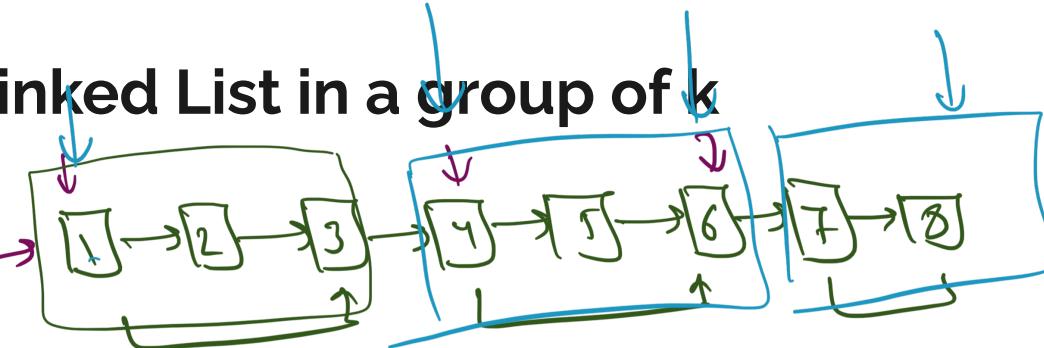
\downarrow^{\uparrow^4}
 $\text{genode}(4)$



Reverse a Linked List in a group of k

K = 3

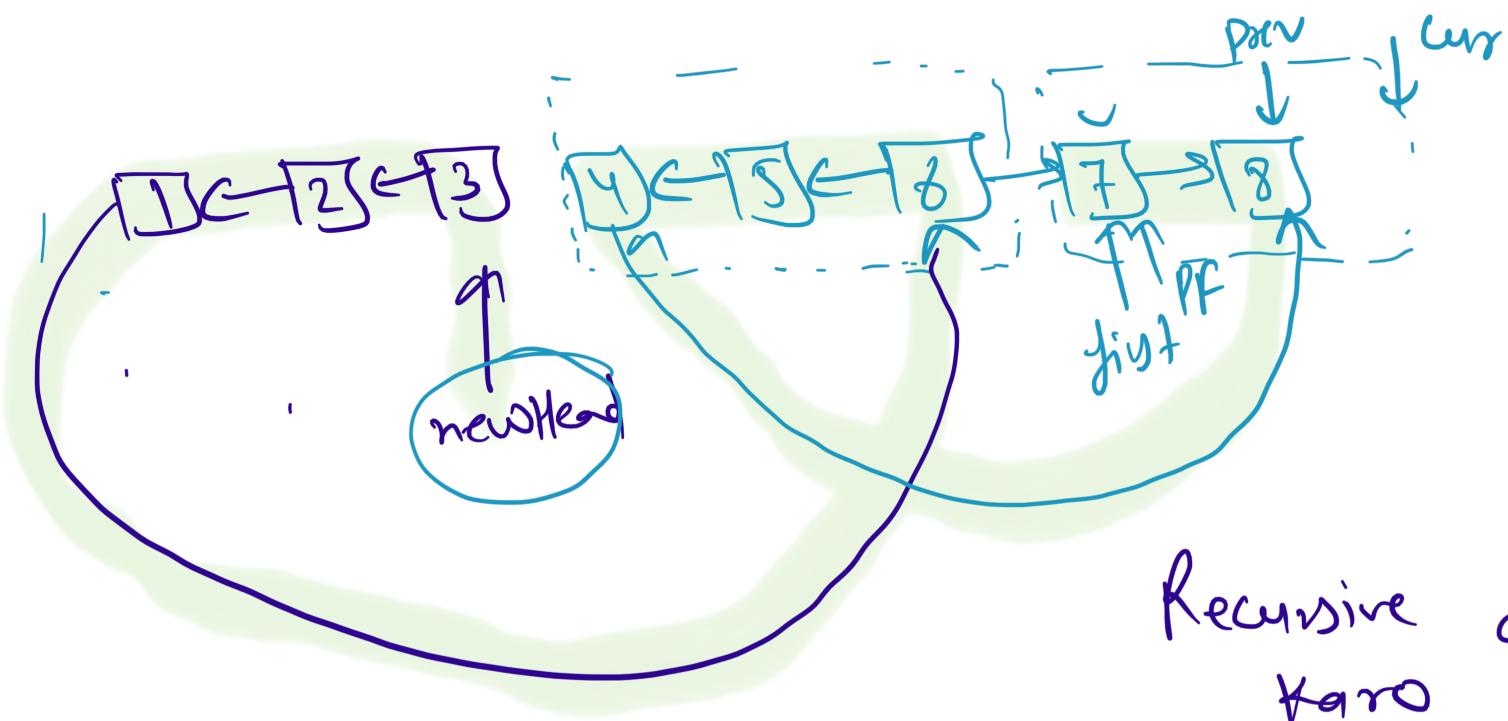
prevfirst



3 → 2 → 1 → 6 → 5 → 4 → 8 → 7

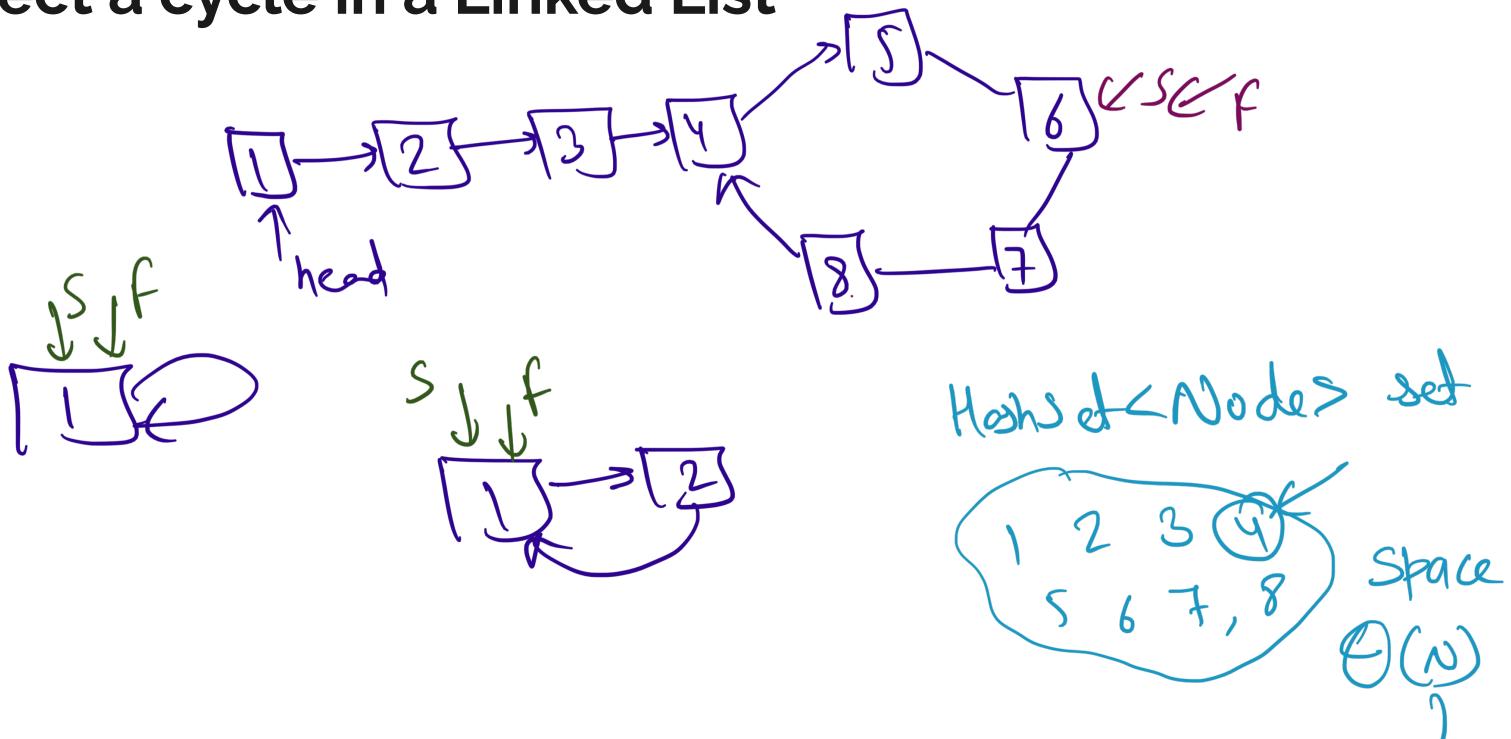
1

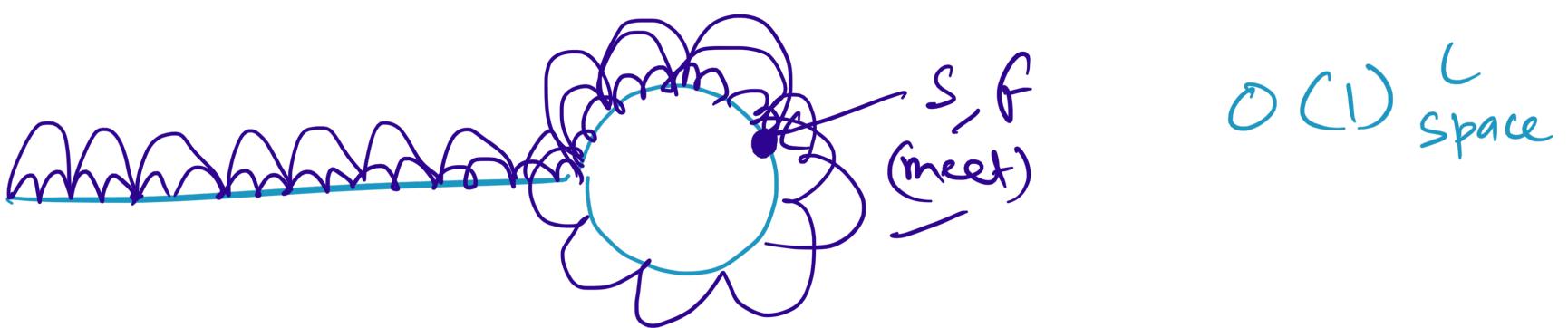
3 → 2 → 1 → 6 → 5 → 4



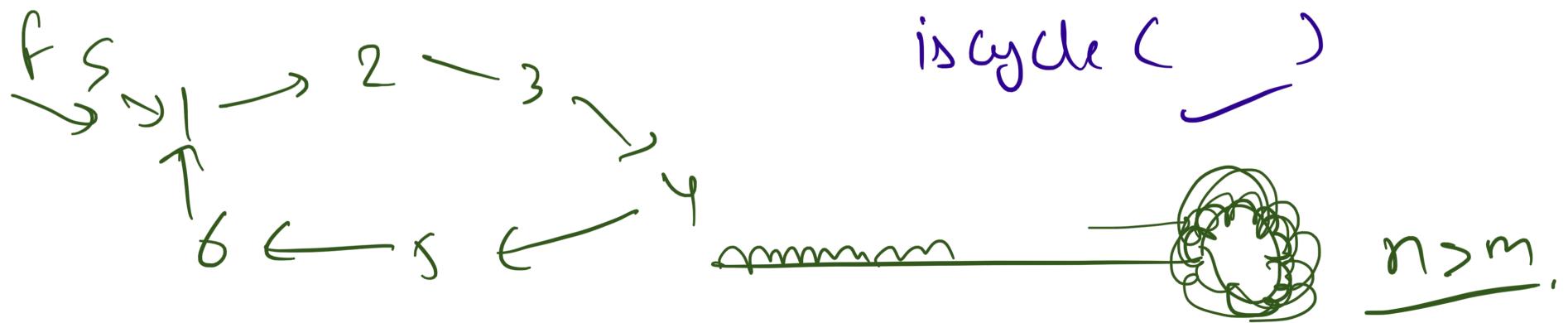
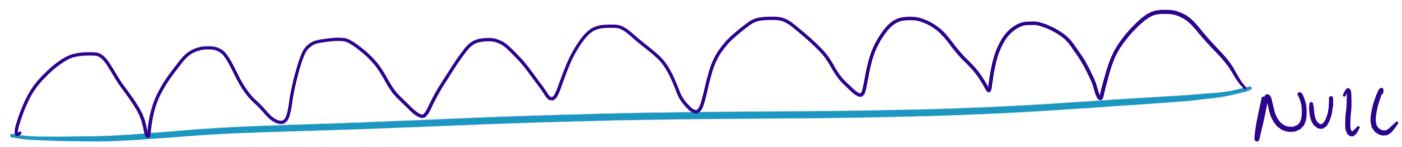
Recursive as H.W.
Karo

Detect a cycle in a Linked List

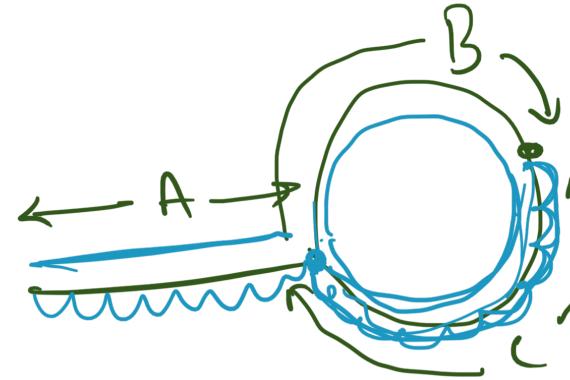
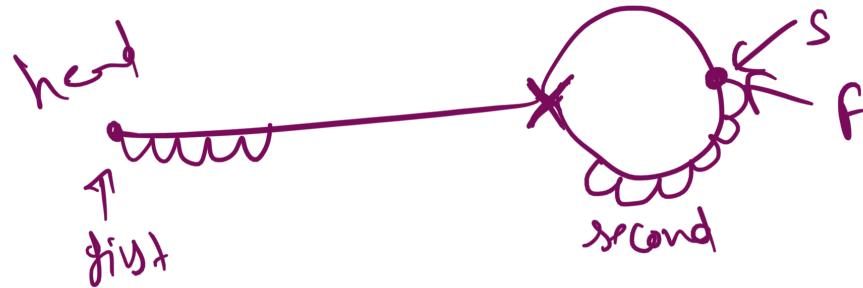




$O(1)$ space



Floyd's cycle detection Algorithm.



$$2 * (\text{dis})_{\text{slow}} = (\text{dis})_{\text{fast}}$$

$$2 * (A + m(B+C) + B) = A + n(B+C) + B$$

$$2A + 2m(B+C) + 2B = A + B + n(B+C)$$

$$\underline{A + B} = \boxed{(n - 2m)(B+C)}$$

$$A = \frac{\lambda(B+C)}{\lambda=0} + C$$

$$\boxed{A = C}$$

$$A = (B+C) + C$$

$$\lambda=1$$

$$(B+C) + C$$

$$\lambda=2$$

$$A = 2(B+C) + C$$

Practice Problems

1. Palindrome Linked List.
2. Rotate a Linked List by k nodes.
3. Add two numbers (each digit is present inside a node of linked list)
4. Merge a linked list into another linked list at alternate positions.