LinkedList

Tuesday, 24 January 2023 11:57 PM

- Find first node in loop in linkedList -> https://www.geeksforgeeks.org/find-first-node-
- Loop or Cycle detection:
    - https://www.geeksforgeeks.org/detect-loop-in-a-linked-list/
    - Floyd loop detection algorithm: https://www.geeksforgeeks.org/floyds-cycle-fin
- https://www.geeksforgeeks.org/write-a-function-to-get-the-intersection-point-of-two
    - Traverse both the lists,
        - Take the difference,
        - Move the pointer of longer list for both pointers to start from same positi
    - Hashset technique
        - Store visited nodes address in hashset
        - Iterate 1 list completely and store its address in hashset
        - When we iterate 2nd list, check the address matches or not
    - Stack pop()/peek() technique from the end
    - 2 pointer technique
        - Move the pointers of both lists parallelly
        - When 1 list completed, point the list to the other linked list and,
        - Iterate again
- Design Browser History
    - Design Browser History | EP 21
        - https://www.tutorialcup.com/leetcode-solutions/design-browser-history-
        - Linked List approach
        - 2 stack approach
- https://www.geeksforgeeks.org/merge-two-sorted-linked-lists/
    - Reverse both linked lists
    - Create the new linked list reference,
    - Start moving the compared bigger node into the new reference, keep building it
    - And finally move the rest of the nodes into the list.

- Flattening a linked list
    - https://takeuforward.org/data-structure/flattening-a-linked-list/

- Flattening a linked list
    - https://takeuforward.org/data-structure/flattening-a-linked-list/