

# Linked List - 1

# In This Lecture

1. Implementation of Linked Lists
2. Insert, Delete, and Iterate in a Linked List
3. Implementation of Doubly Linked List

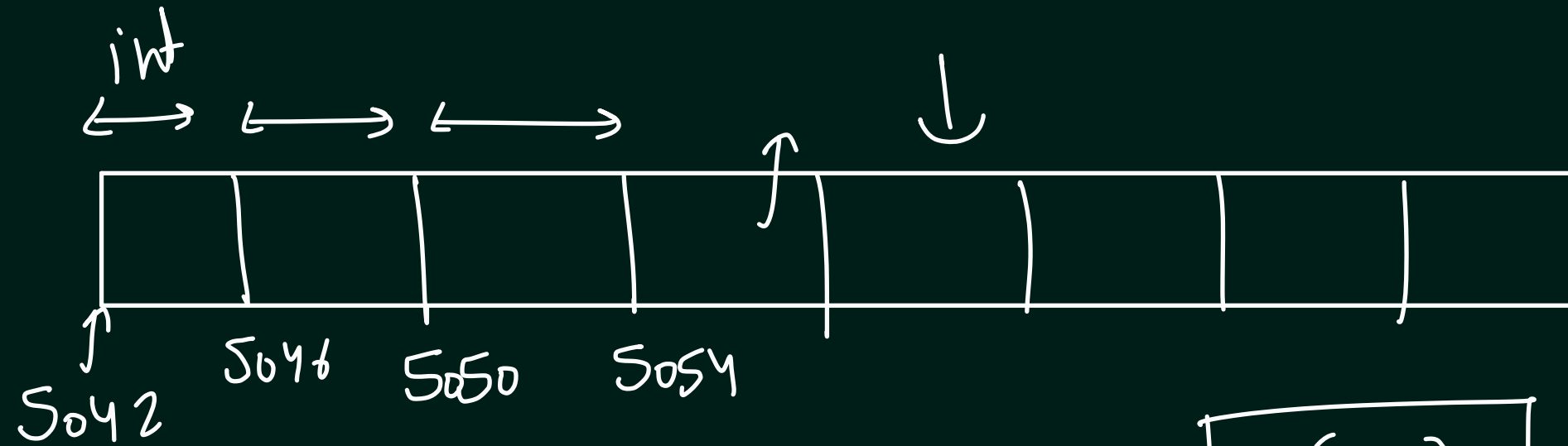
# Implementation of Linked Lists

It is a Linear Data Structure where elements are not stored at contiguous memory locations, however as nodes of a LinkedList are connected, it is treated as a Linear Data Structure. A linked list is a collection of nodes where every node contains two fields i.e. data, address fields. The Data field contains the actual value of the node while the address field contains the address of the next node.



# Insert, Delete, and Iterate in a Linked List

Contiguous



$10^6$

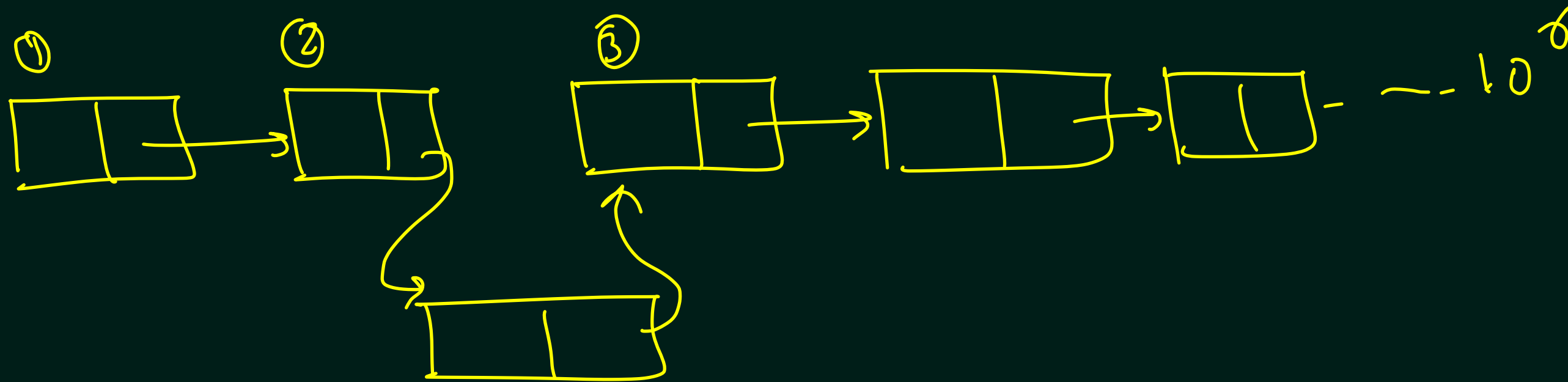
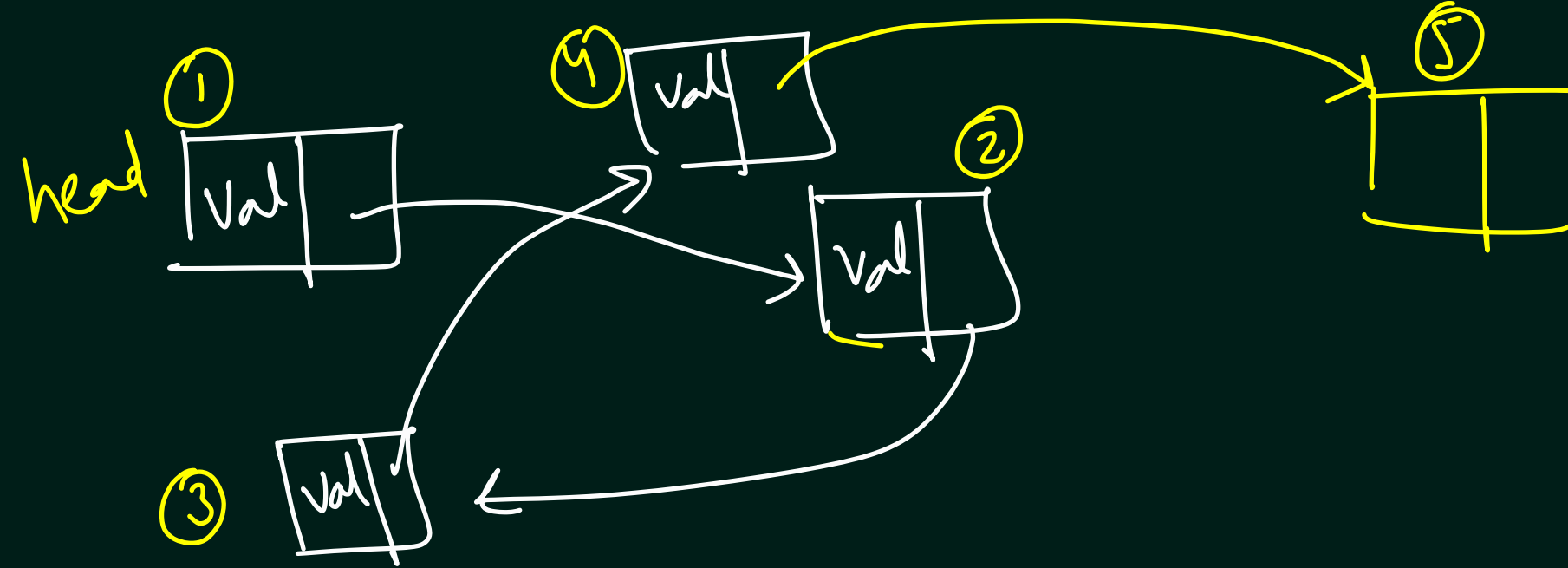
$O(n)$

$$5042 + 3 \times 4$$

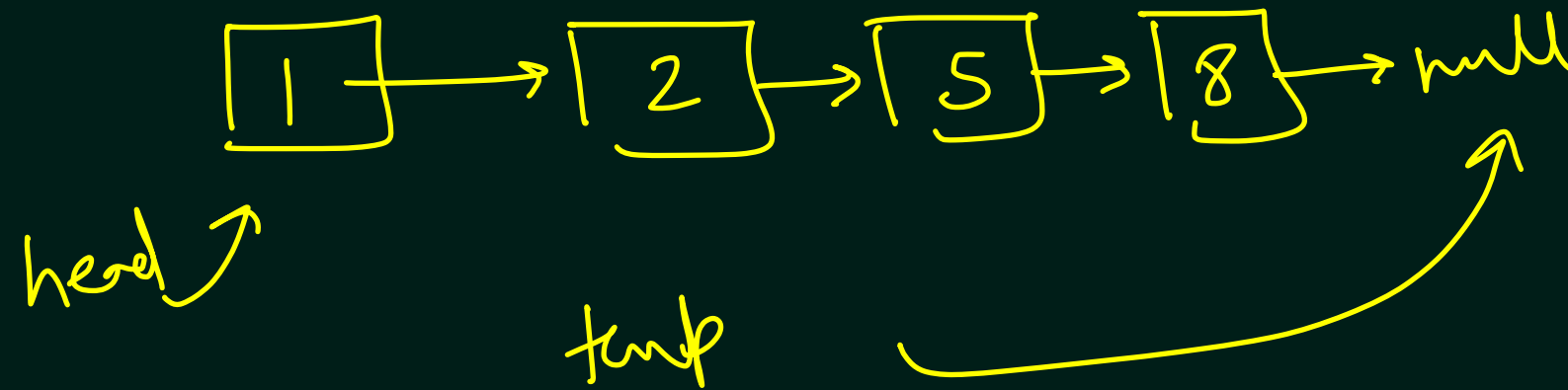
- ① Insert and delete take time
- ② Specify the size before creating array.

# Insert, Delete, and Iterate in a Linked List

Non  
Contiguous

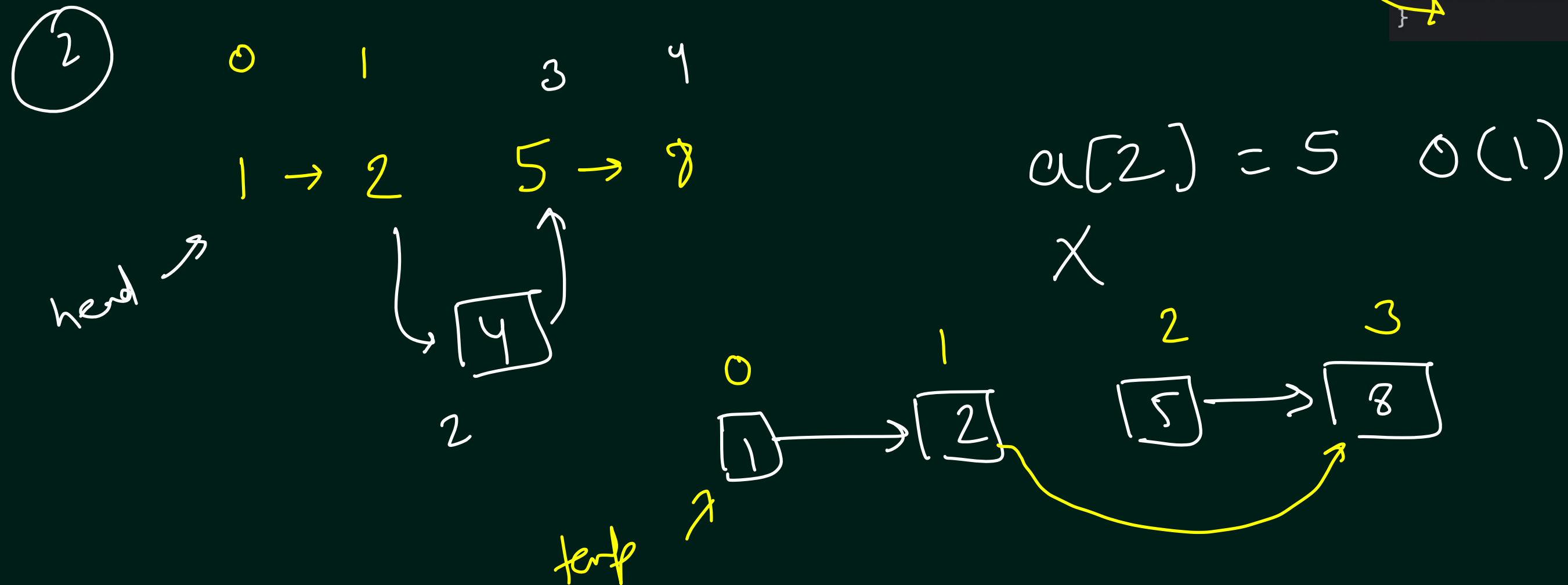


# Insert, Delete, and Iterate in a Linked List

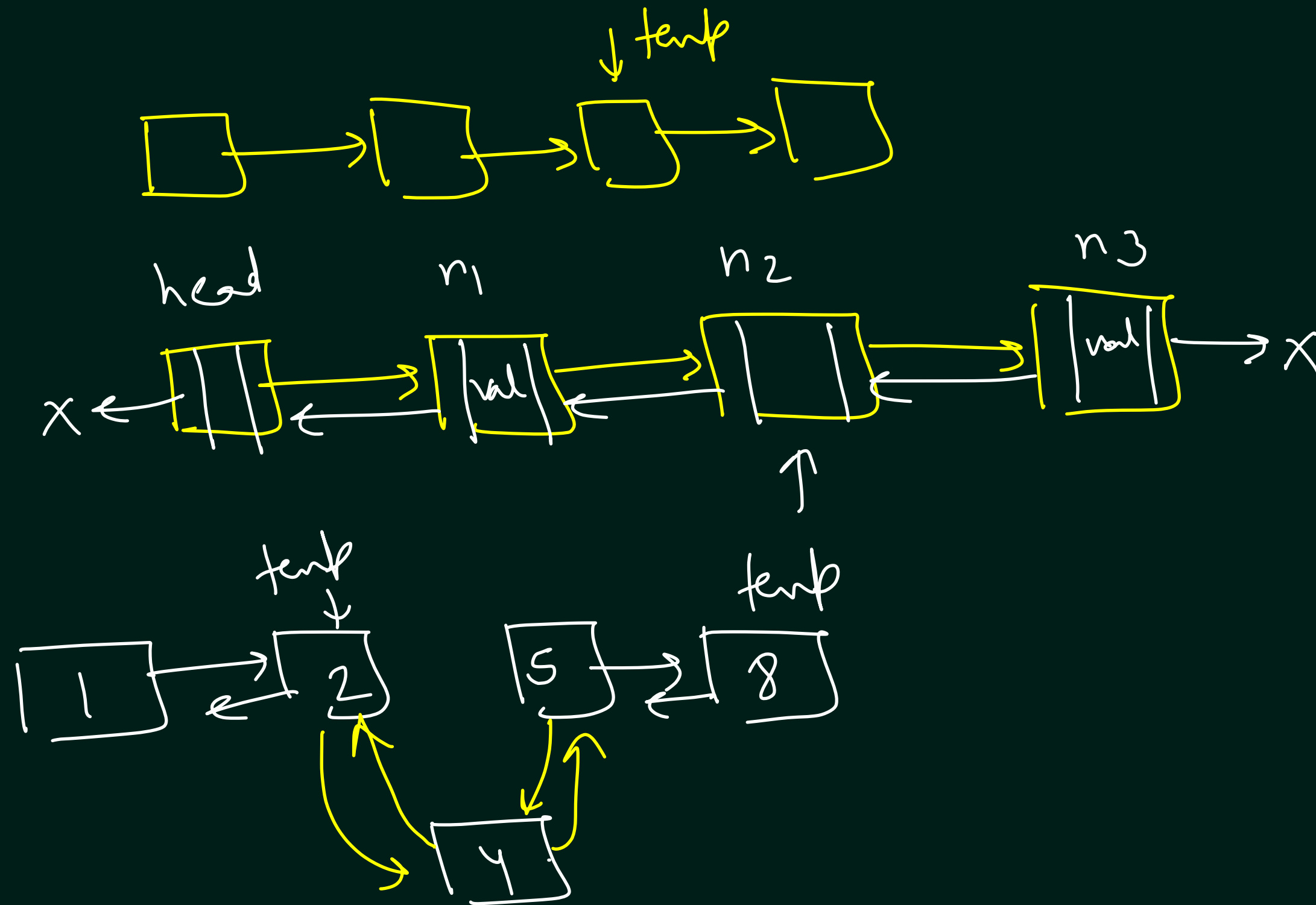


```

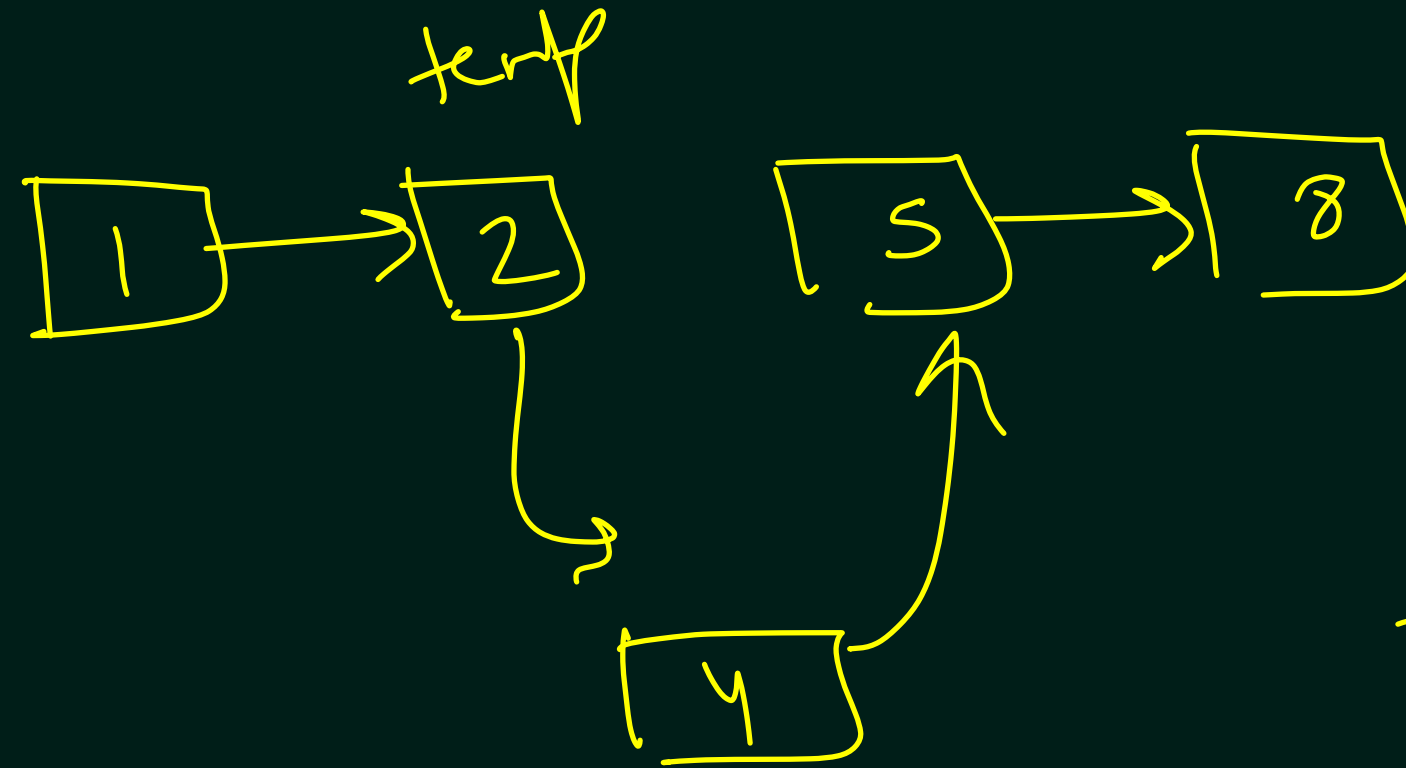
public static void printLinkedList(Node<Integer> head) {
    Node<Integer> temp = head;
    while (temp != null) {
        System.out.print(temp.val+" -> ");
        temp = temp.next;
    }
    System.out.println();
}
  
```



# Insert, Delete, and Iterate in a Doubly Linked List



# Insert, Delete, and Iterate in a Doubly Linked List



`newNode.next = temp.next`  
`temp.next = newNode`