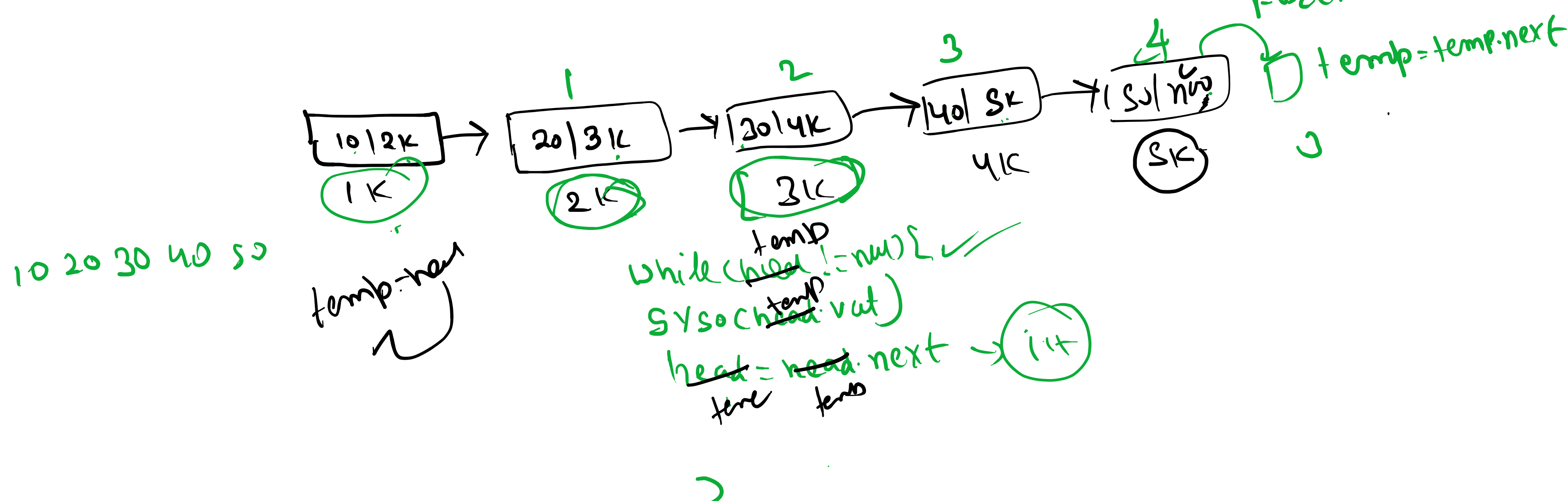
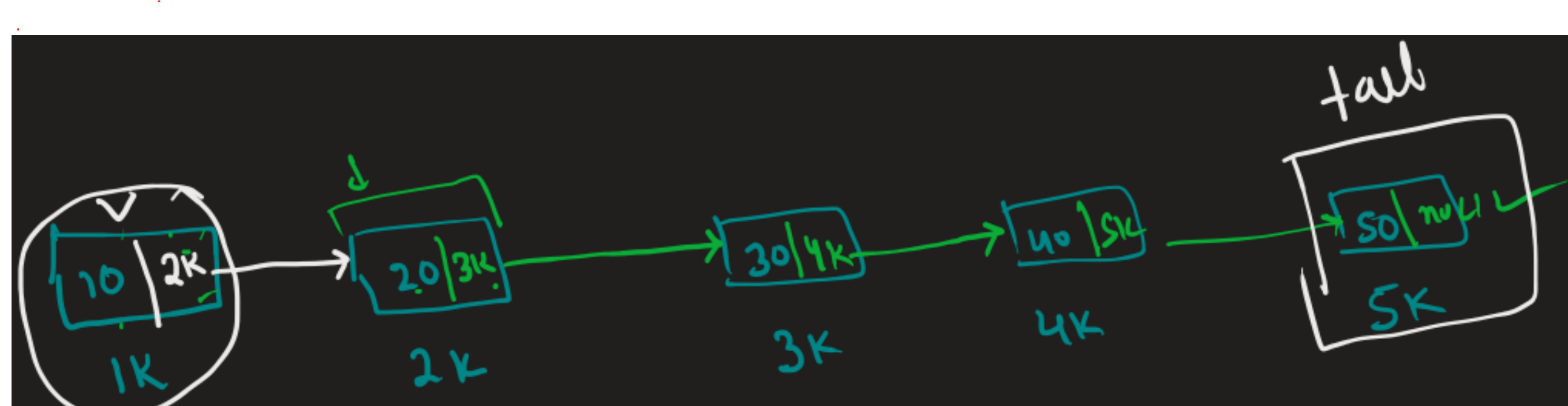
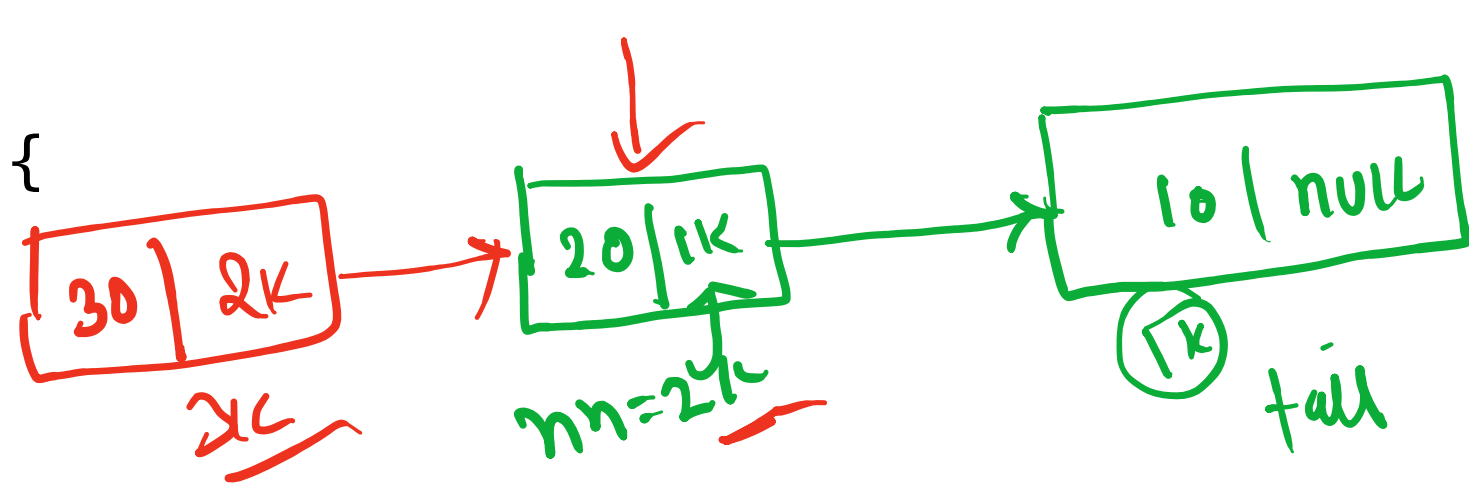
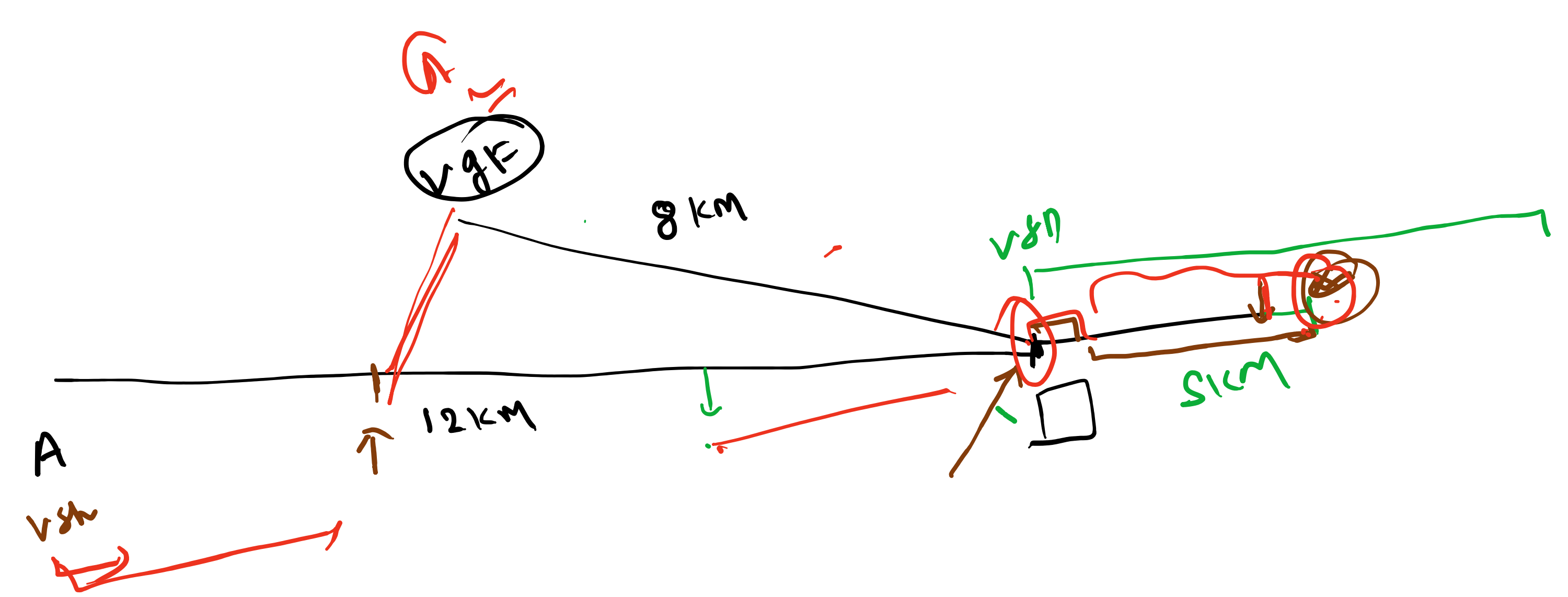
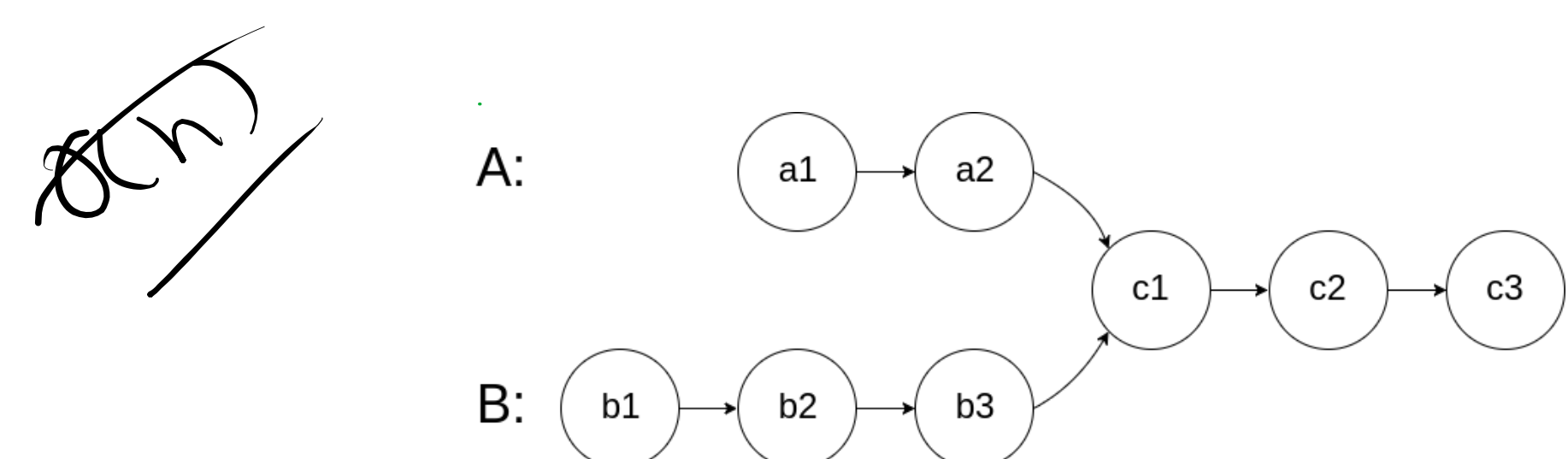
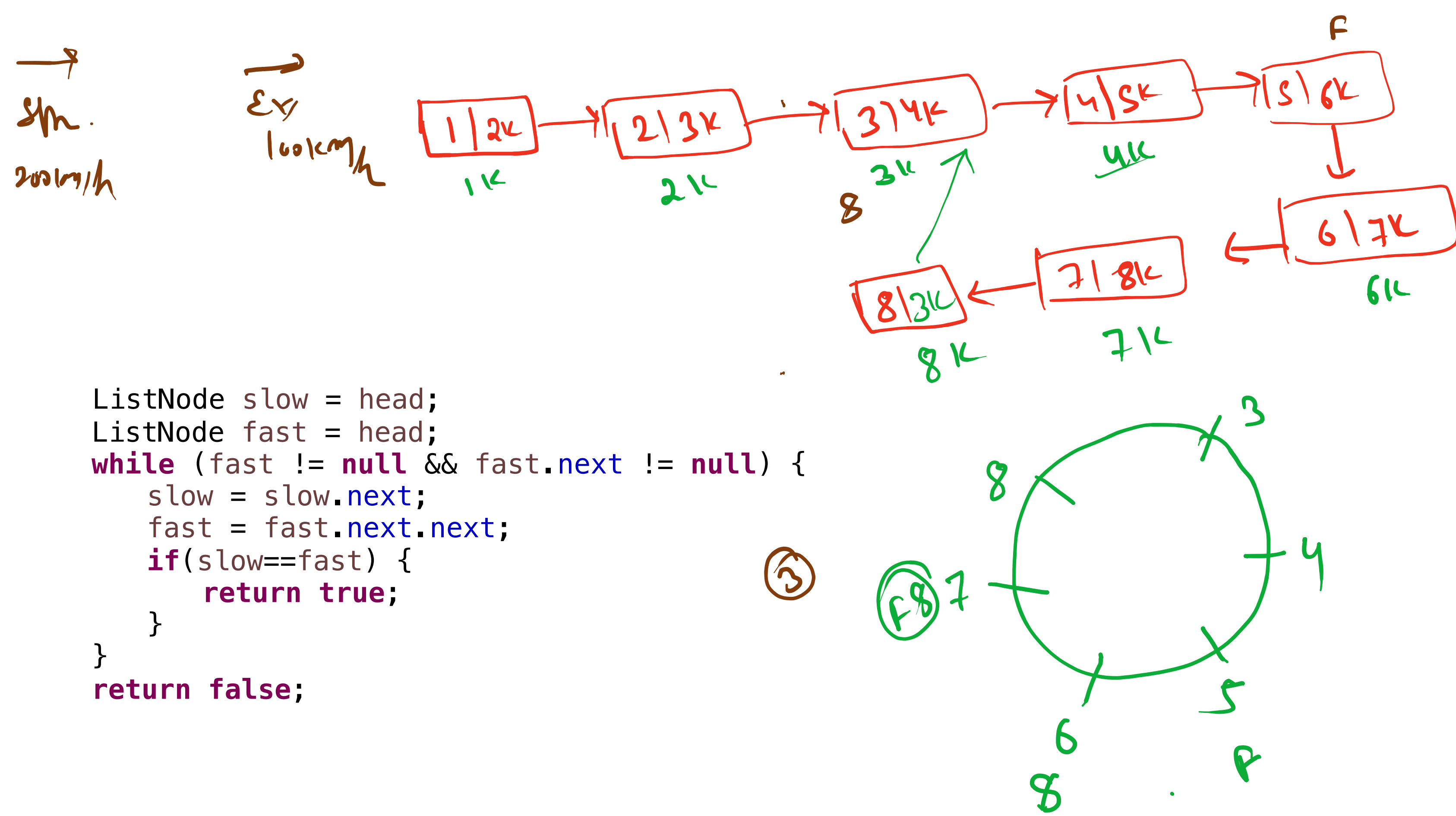
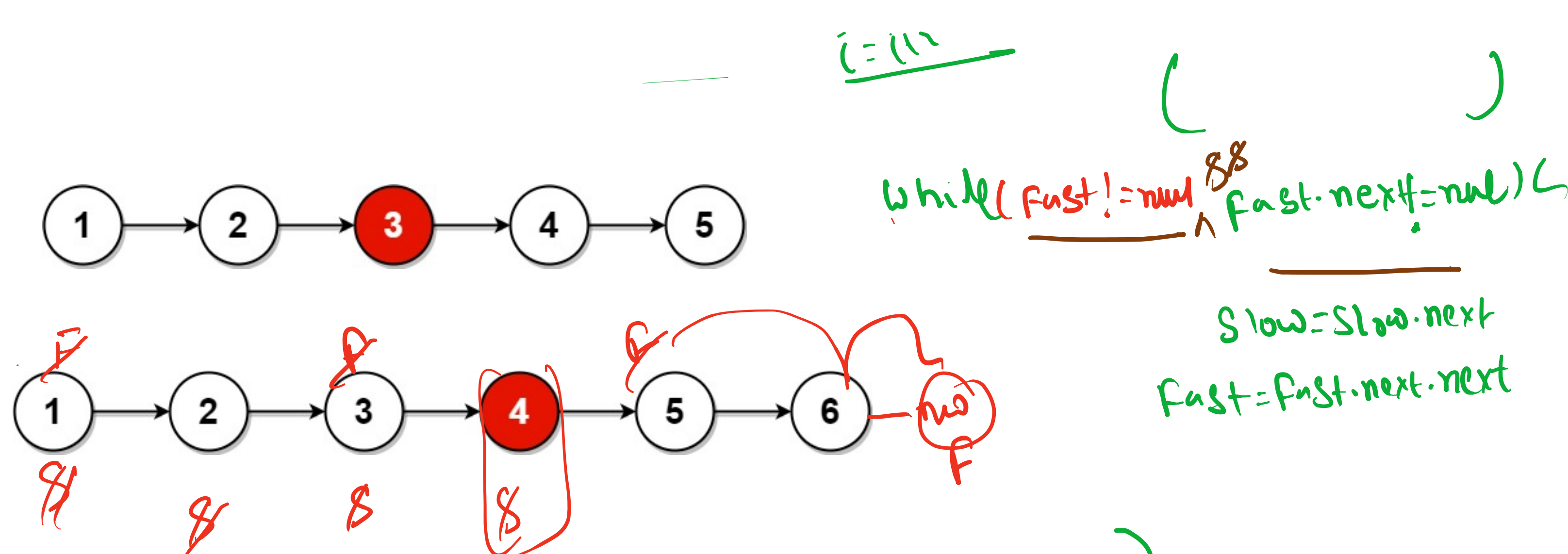
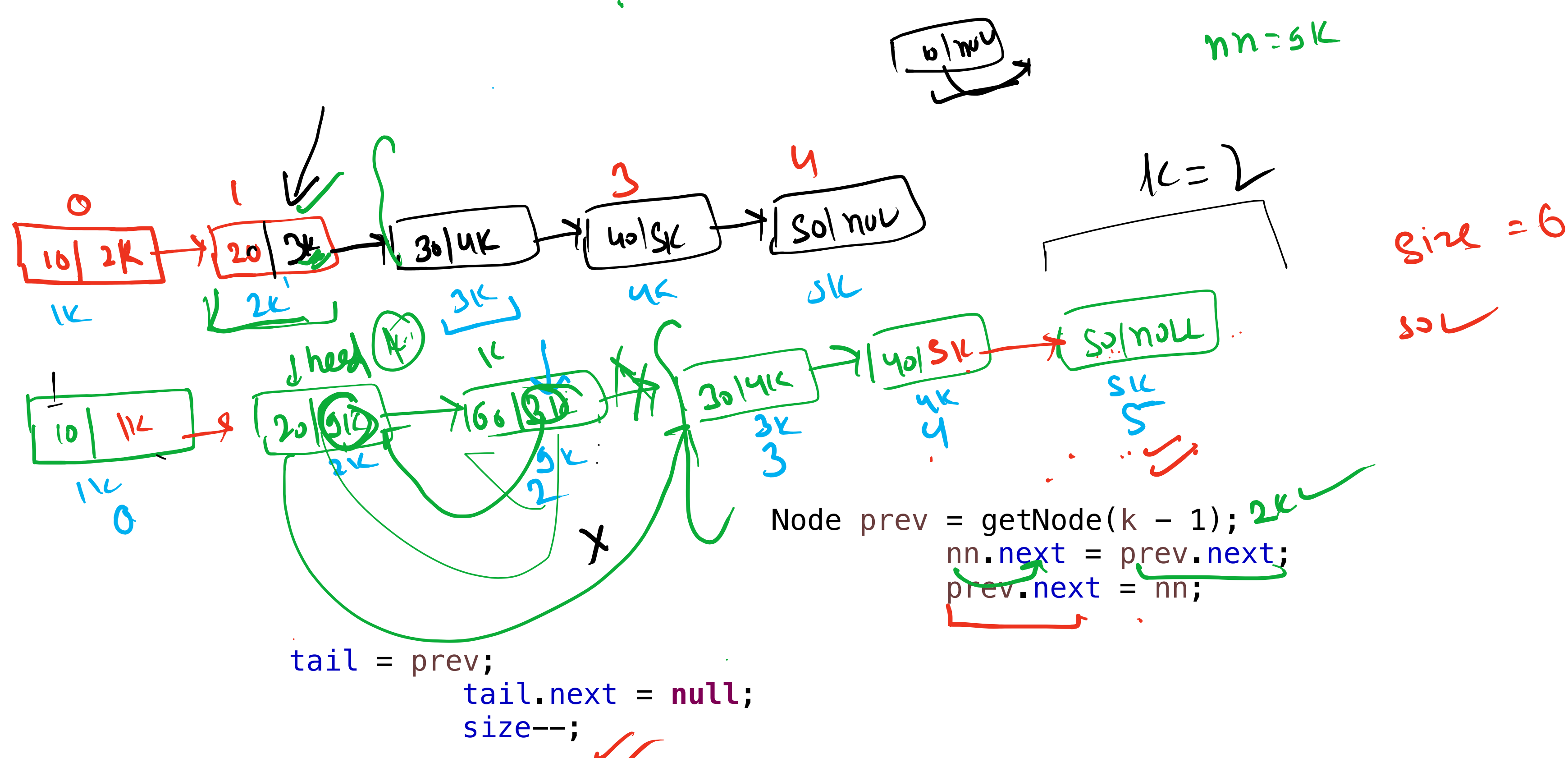


```
public void addFirst(int item) {  
    Node nn = new Node();  
    nn.val = item;  
    if (size == 0) {  
        head = nn;  
        tail = nn;  
        size++;  
    }  
}
```



```
public int removeFirst() {  
    Node temp = head;  
    if (size == 1) {  
        head = null;  
        tail = null;  
        size--;  
    } else {  
        head = head.next;  
        temp.next = null;  
        size--;  
    }  
    return temp.val;  
}
```



```
public ListNode getIntersectionNode(ListNode headA, ListNode headB) {  
    ListNode a = headA;  
    ListNode b = headB;  
    while (a != b) {  
        if (a == null) {  
            a = headB;  
        } else {  
            a = a.next;  
        }  
        if (b == null) {  
            b = headA;  
        } else {  
            b = b.next;  
        }  
    }  
    return a;  
}
```

