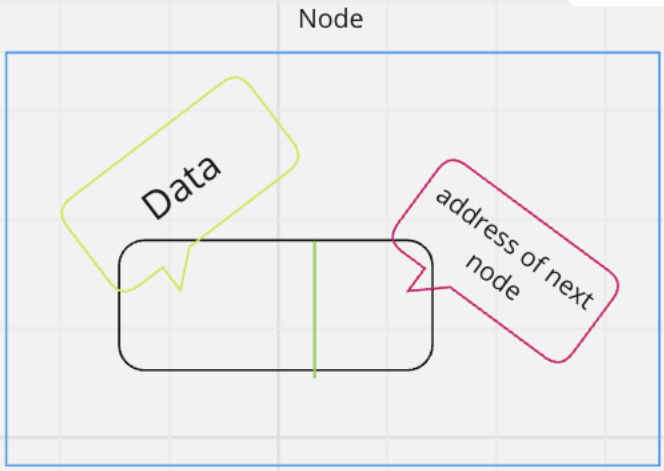
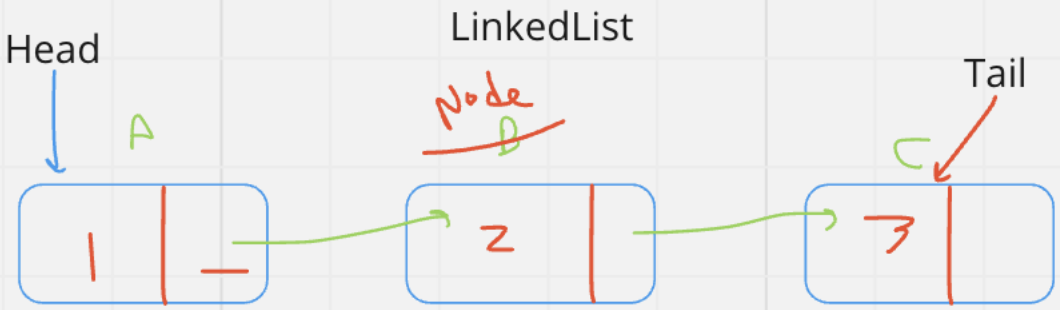
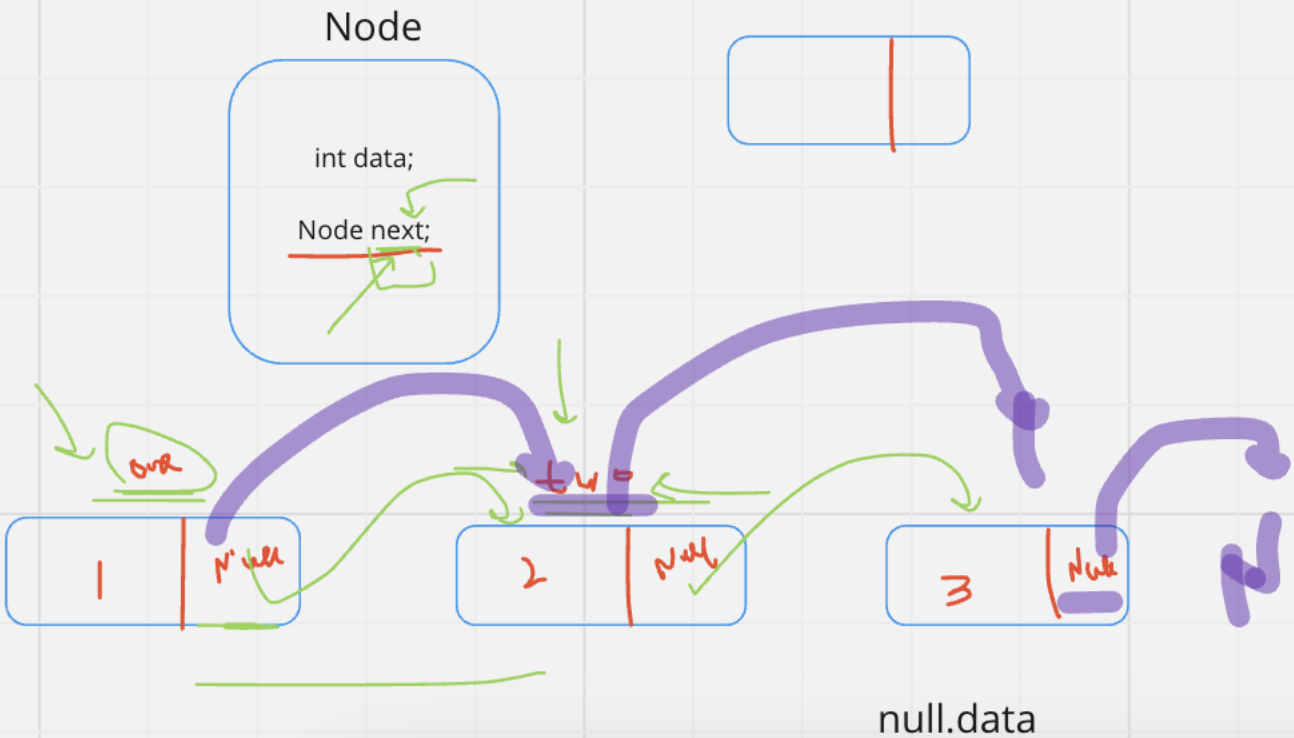
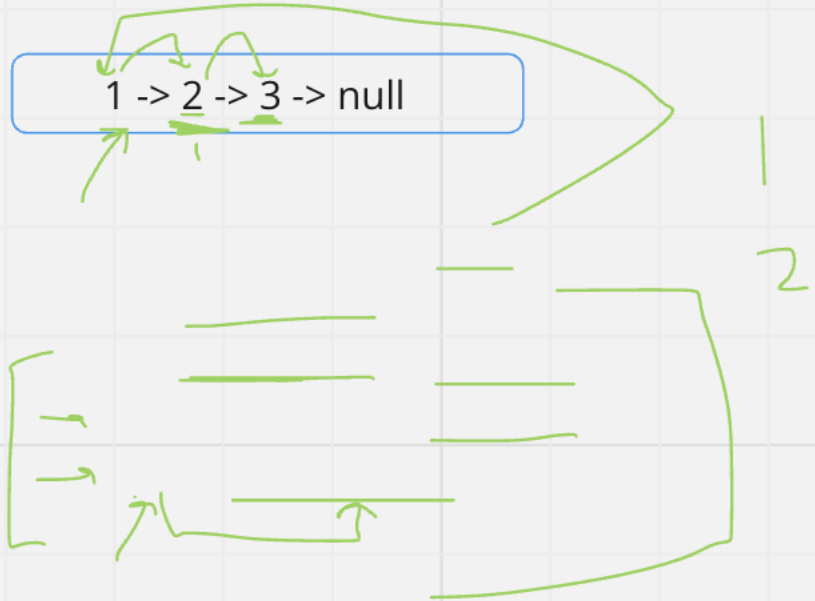

Node
Data
address of next node

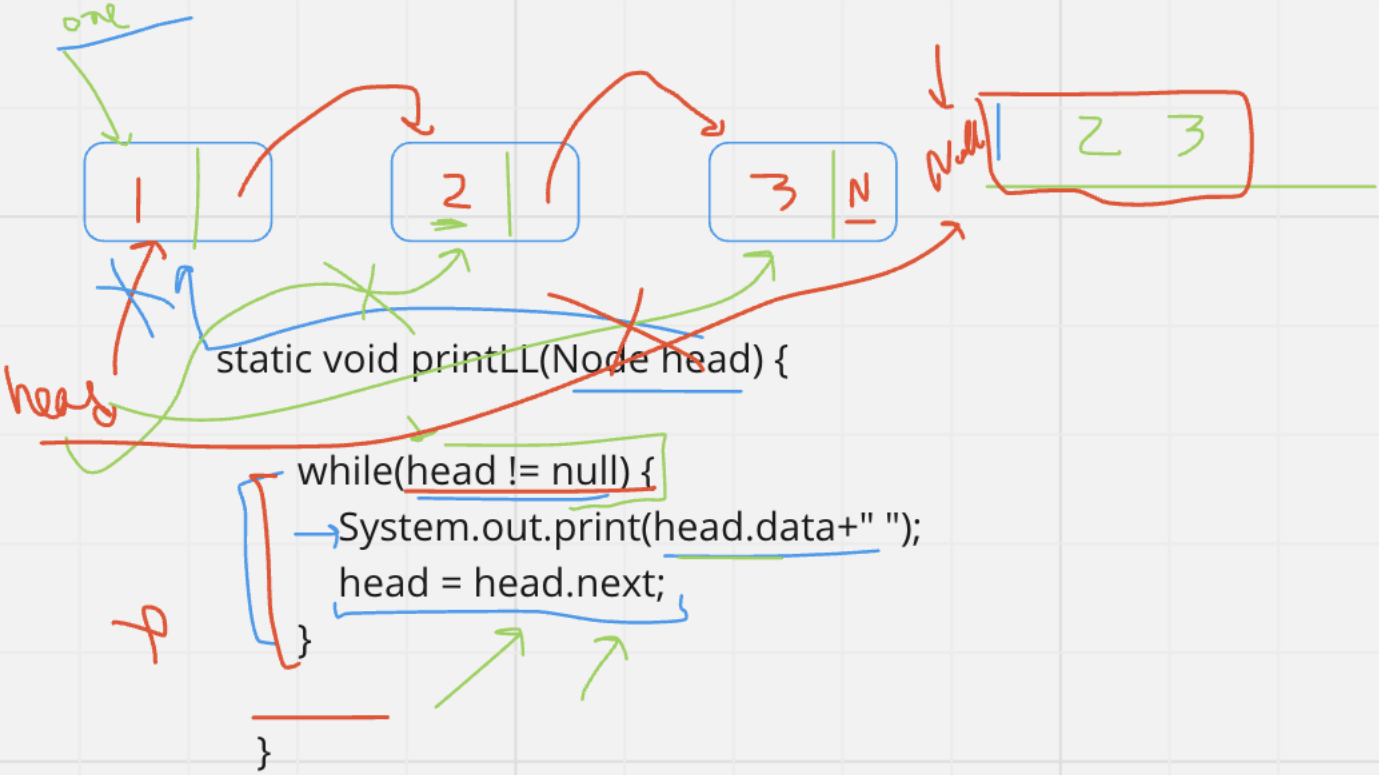

Head
LinkedList
Tail
A
Node
B
C
1
2
3

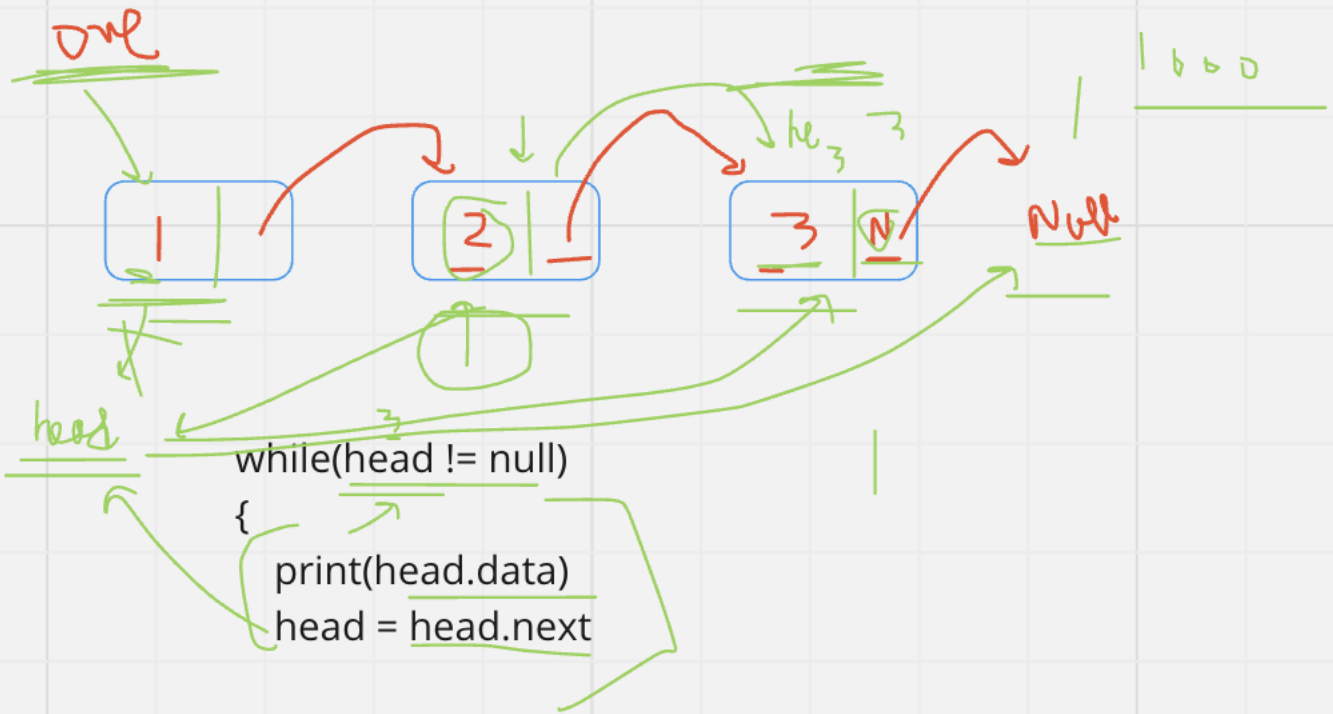

Node
int data;
Node next;
1
2
3
null.data

one.data => 1

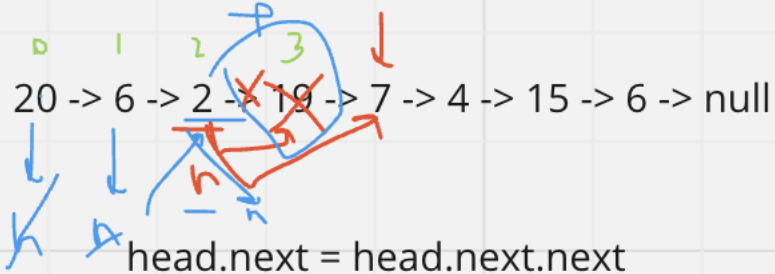
one.next.next.next.data => 2




one
1
2
3
N
Null
2 3
head
static void printLL(Node head) {
while(head != null) {
System.out.print(head.data+" ");
head = head.next;
}
}


1
2
3
Null
while(head != null)
{
print(head.data)
head = head.next
}

20
6
2
19
7
4
15
9



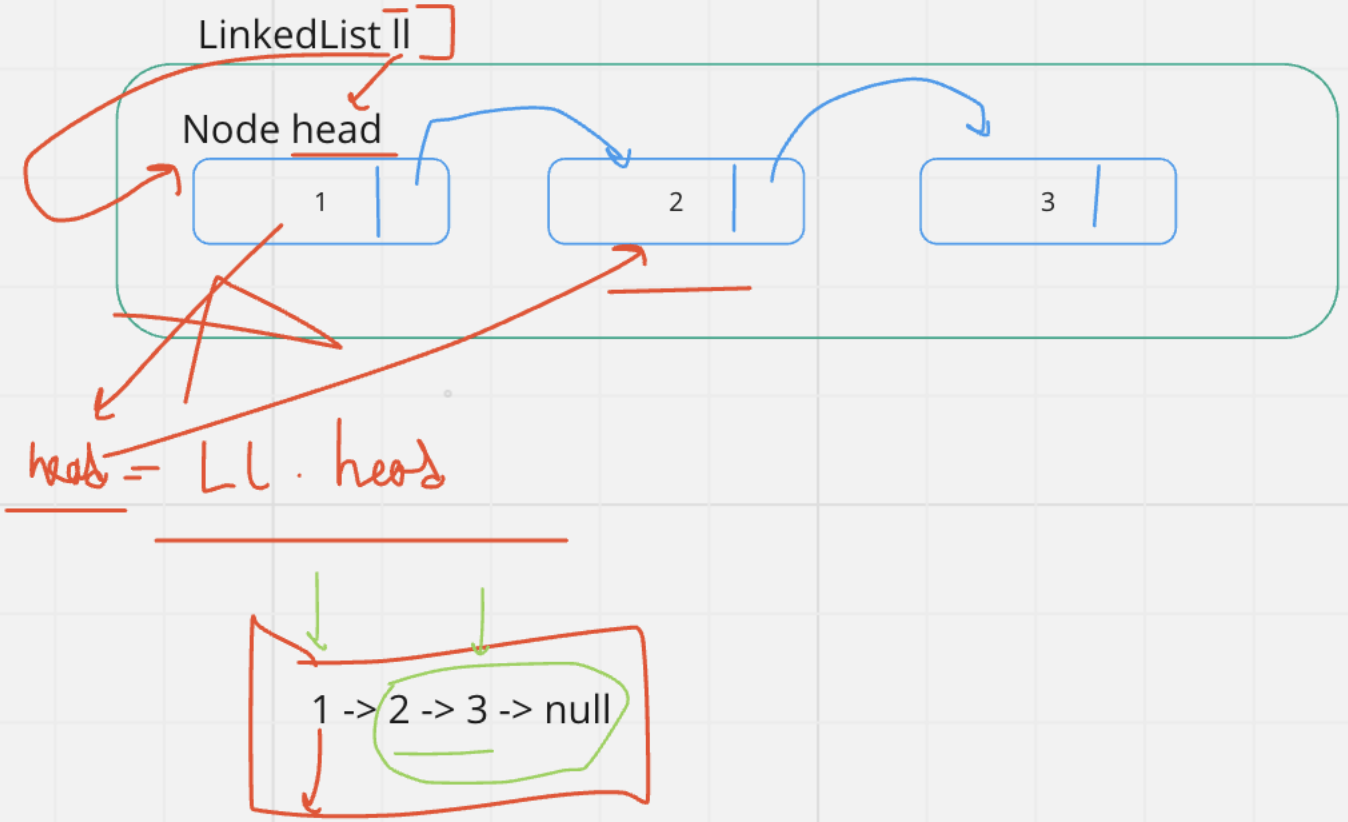
3

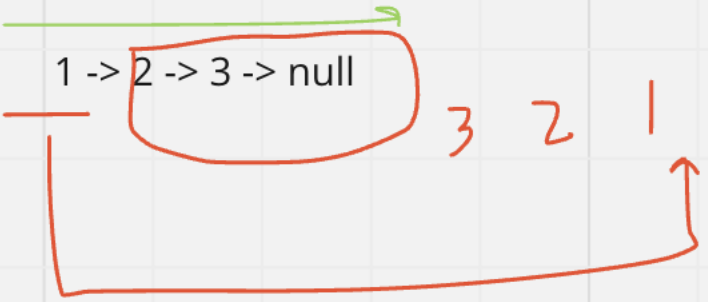
20 -> 6 -> 2 -> 7 -> 4 -> 15 -> 6 -> null

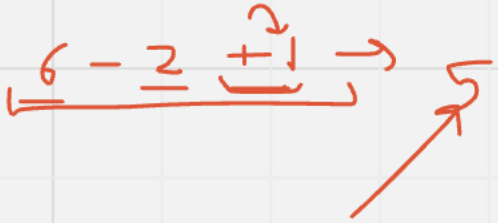
1->2->3

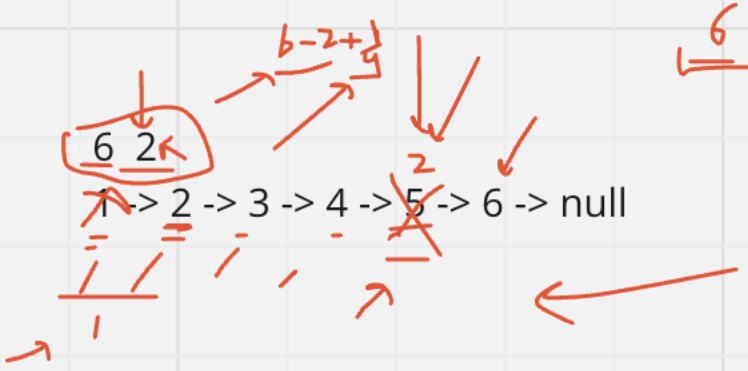
0
head=head.next;

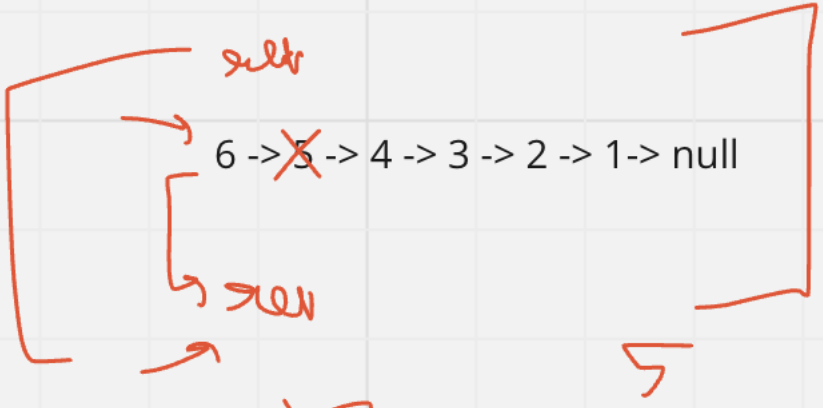
```
Node head = ll.head;

        if(toRemove == 0) {
            ll.head = head.next;
            return;
        }
```

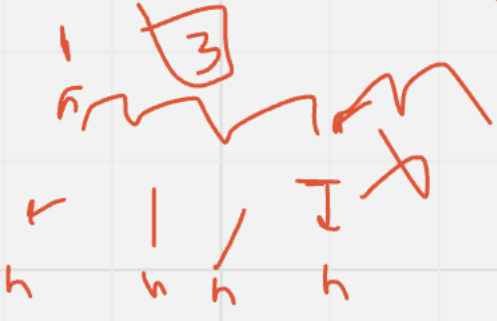


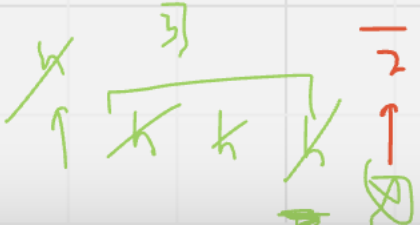

1 -> 2 -> 3 -> null
3 2 1

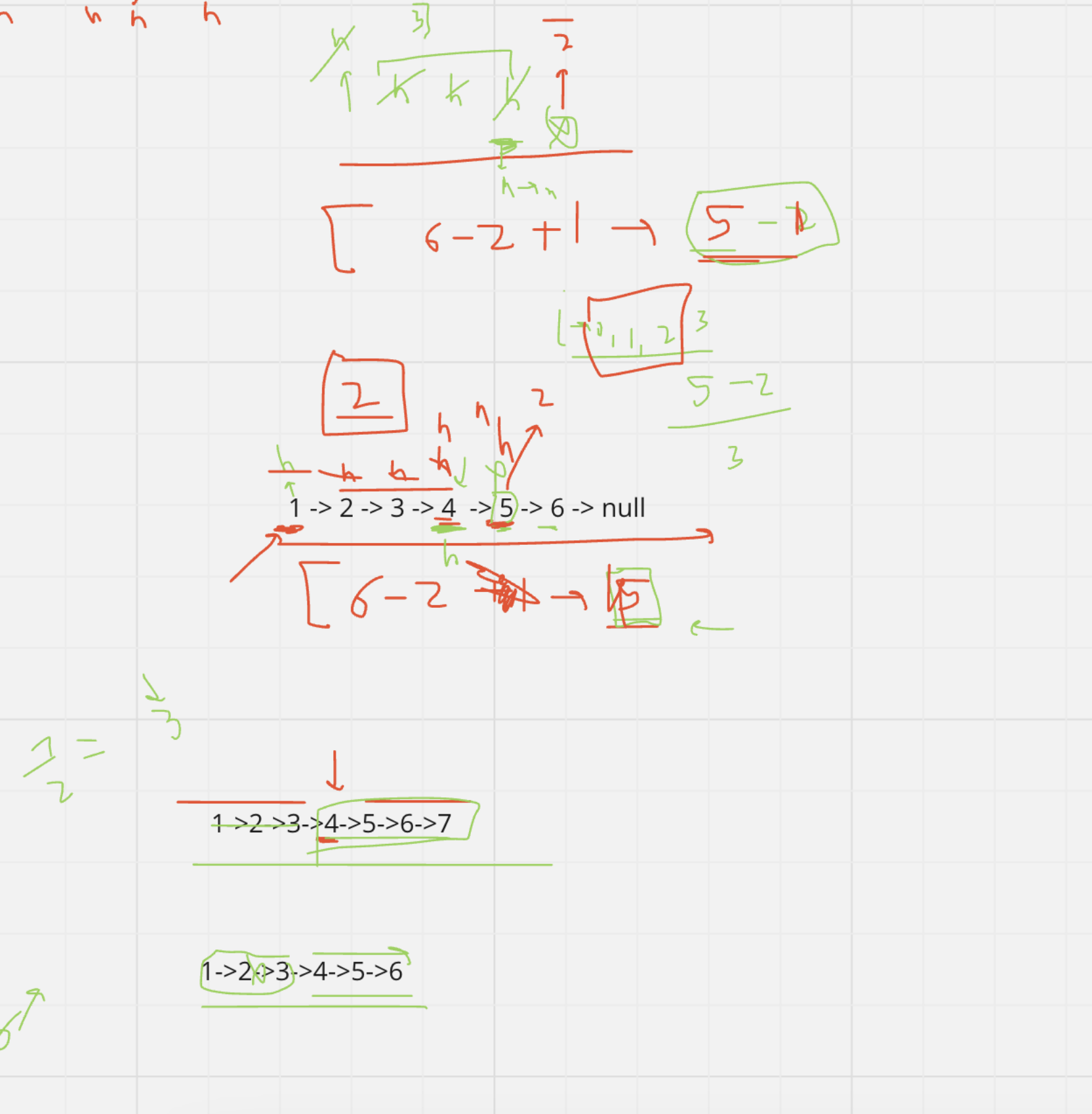

6 - 2 +1


6-2+1
6 2
2
1 -> 2 -> 3 -> 4 -> 5 -> 6 -> null


6 -> 5 -> 4 -> 3 -> 2 -> 1-> null

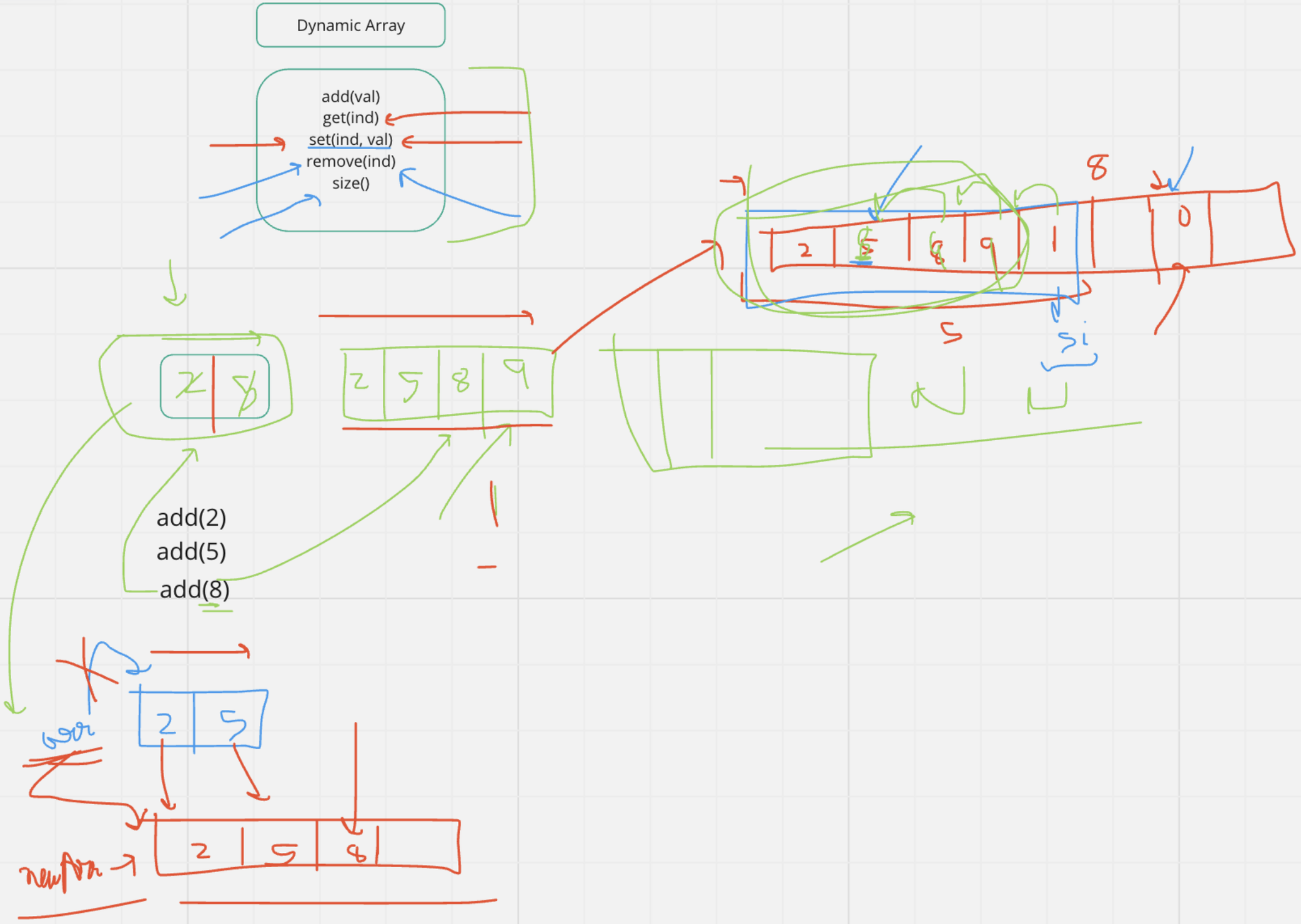

3





1->2->3->4->5->6->7

1->2->3->4->5->6


Dynamic Array
add(val)
get(ind)
set(ind, val)
remove(ind)
size()
add(2)
add(5)
add(8)