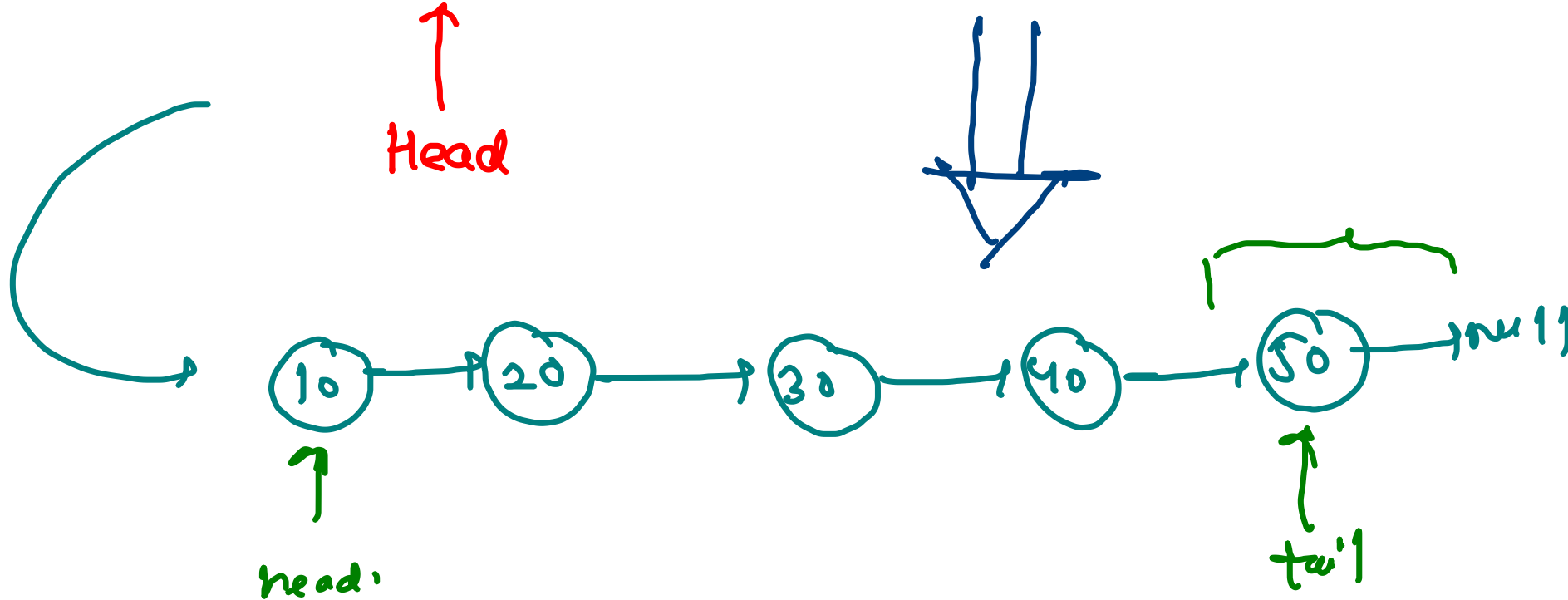
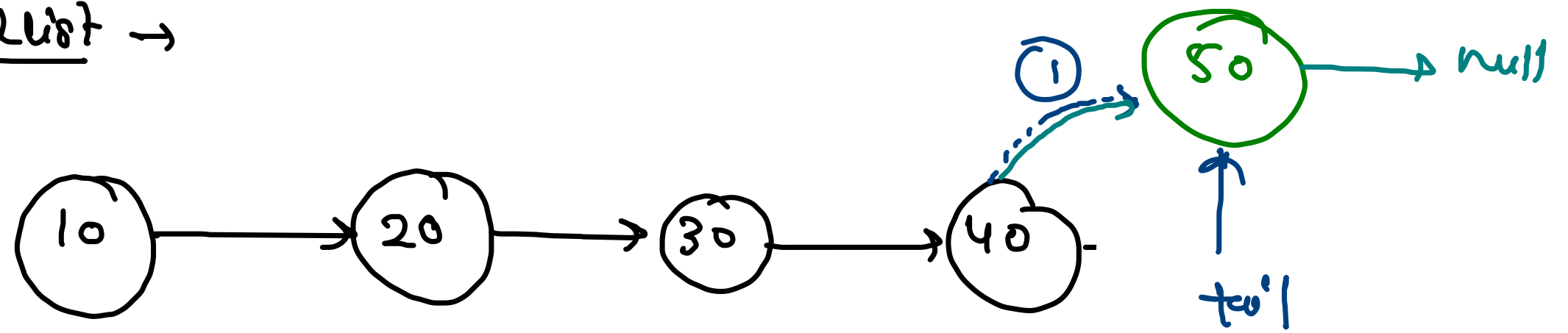




node - class

LinkedList →



connection ✓  
tail.next = nn;  
set tail ✓  
tail = nn

Head x  
tail ✓  
size → size++

```

public class Main {
    public class Person {
        public String name;
        public int age;
        public double ht;

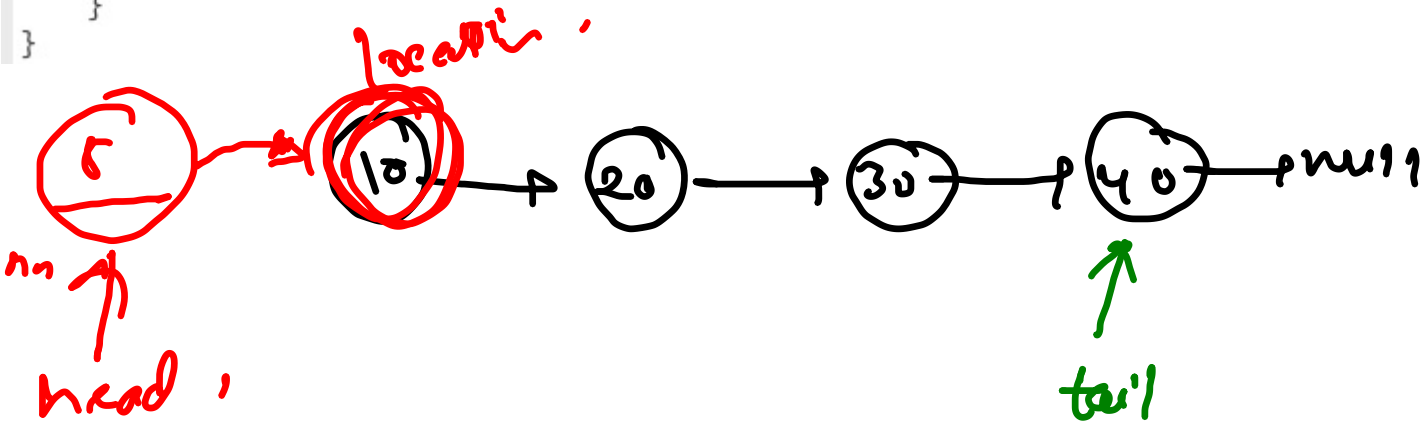
        public Person(String name, int age, double ht) {
            this.name = name;
            this.age = age;
            this.ht = ht;
        }

        public void printDetails() {
            System.out.println(this.name + " " + this.age + " " + this.ht);
        }

        public static void main(String[] args) {
            Person p = new Person("A", 22, 5.9);
            p.printDetails();
        }
    }
}

```

A object →  
 single time Definition Store

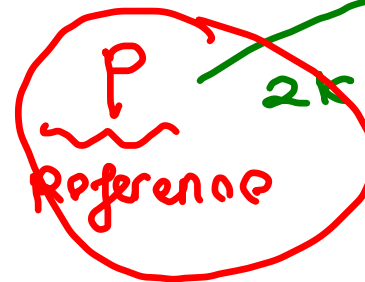


nn.next = head;  
 head = nn;

Size

Size of list

main



Heap

Static segment,  
 + def's  
 + const

String

int

A, 22,

5.9

deg. of method

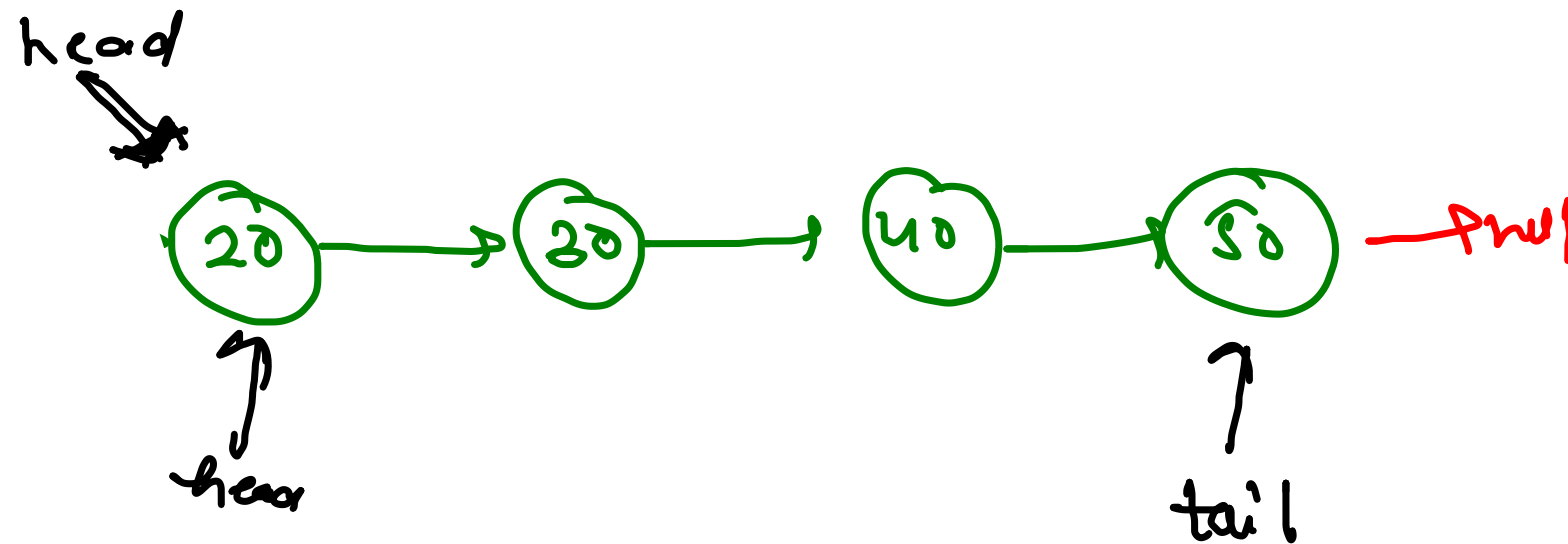
deg. of method

2K

Person class's

object

## Remove First

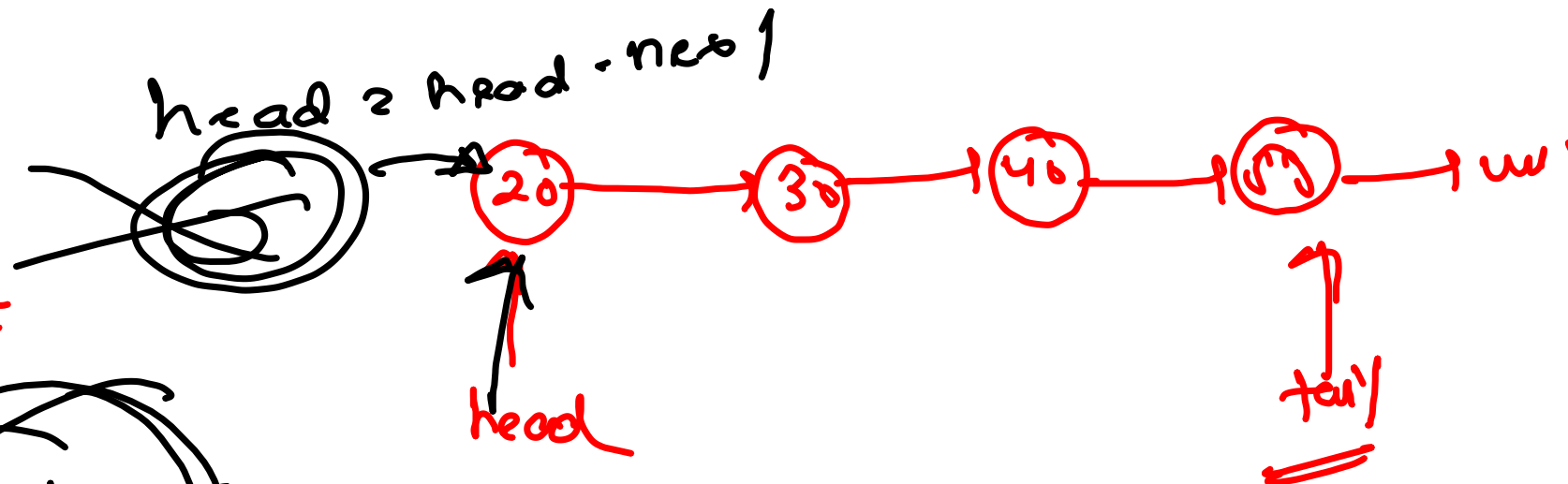


Edge Case =

size = 0

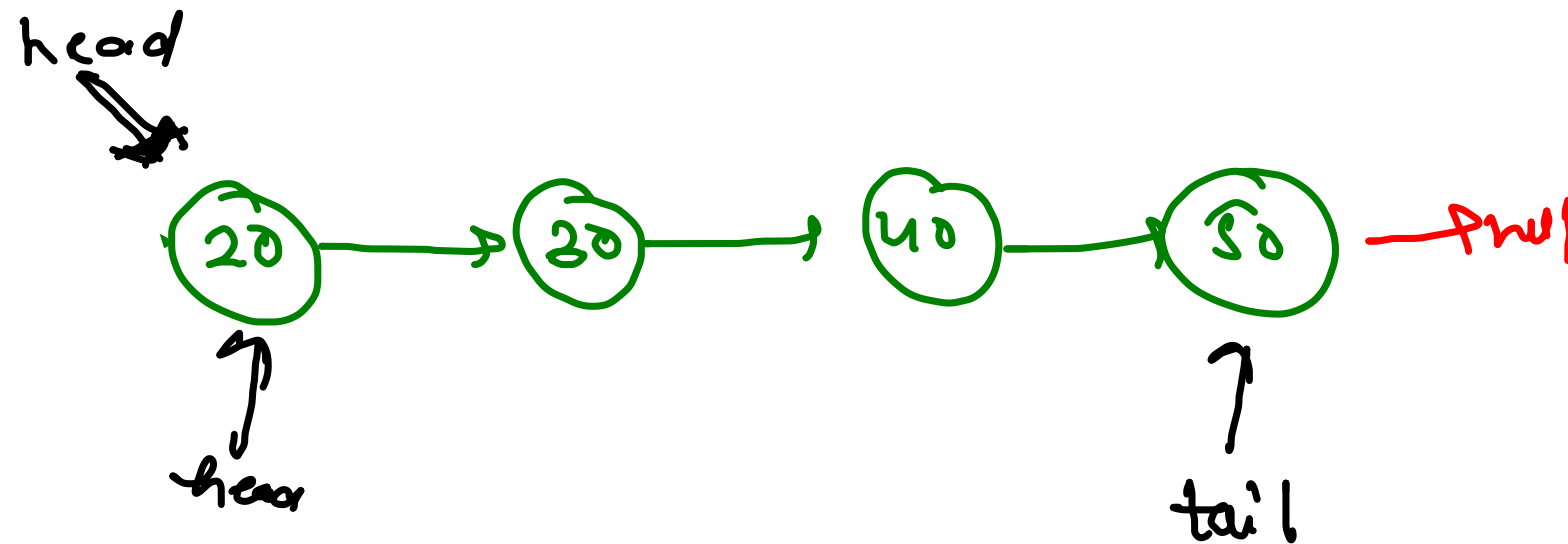
size = 1

After  
remove first



head & tail = null

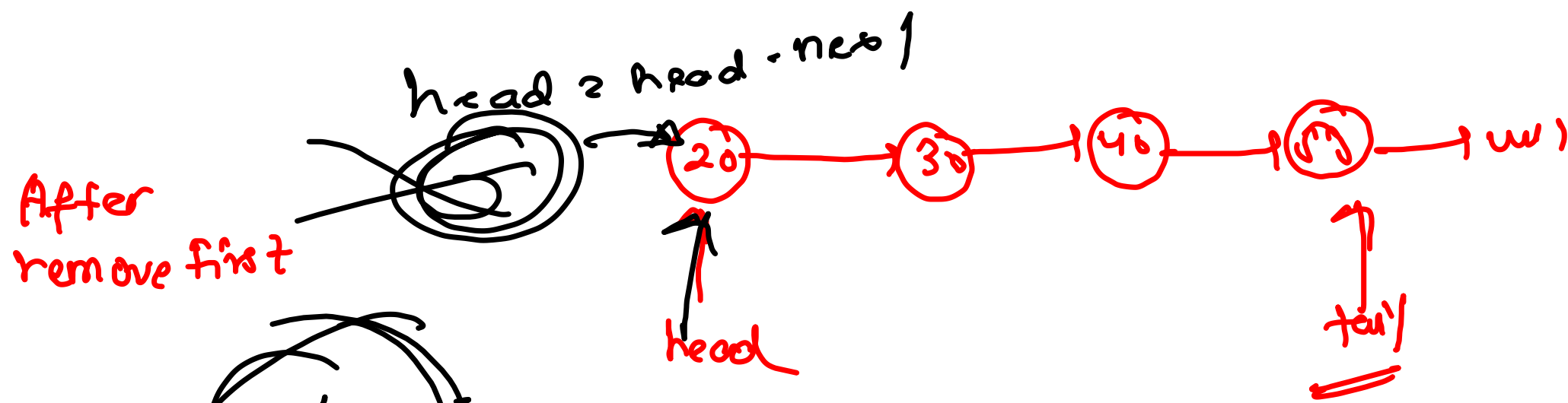
## Remove First



Edge Case =

size = 0

size = 1



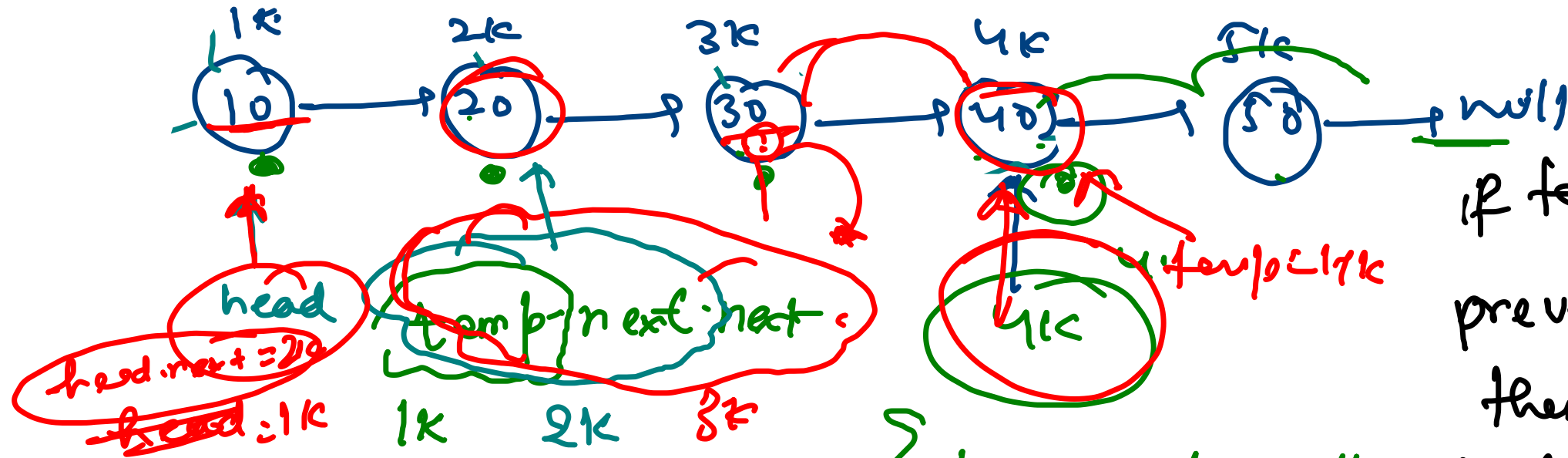
# Remove last

Size == 0

'List is Empty'

Size == 1

head = tail = null;



if tail have previous access then remove last is easy

tail = tail.prev;  
tail.next = null;

Size > 1

temp = 1k

Remove last

temp = temp.next  
2k

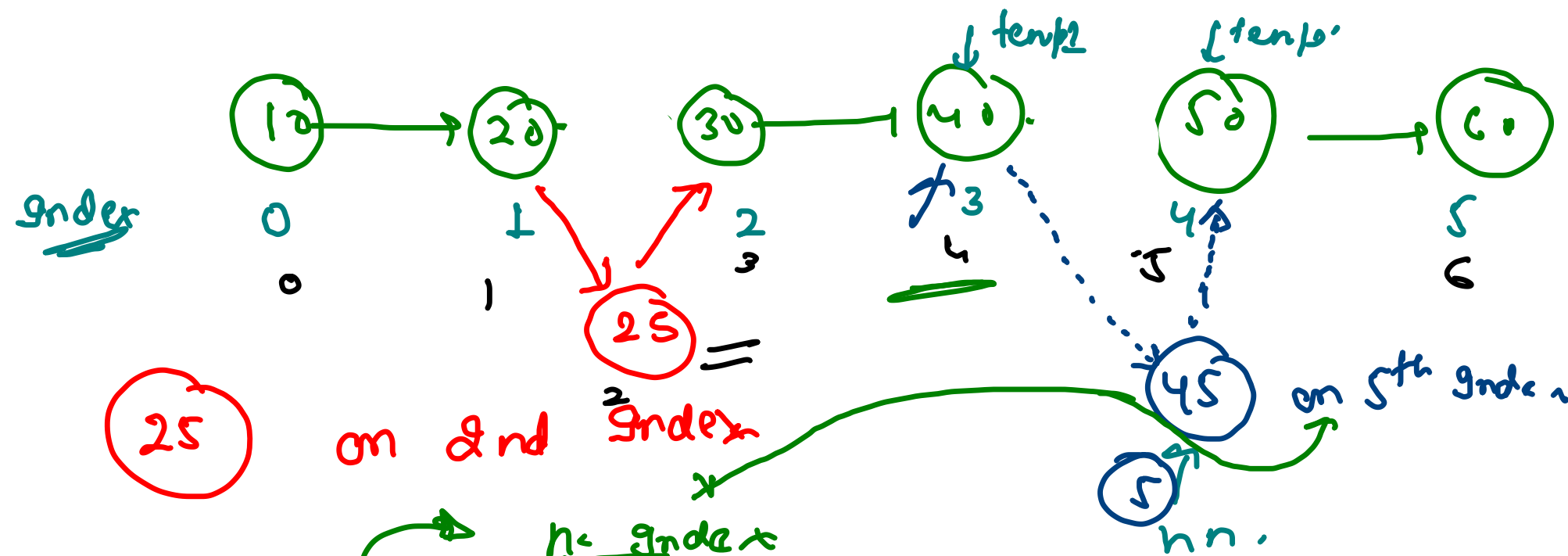
Node temp = head;

temp = temp.next

temp = 4k

move till  
temp.next.next != null  
return temp;

Get Nth node } Abstract  
→ Get second last Node



is appropriate,

if i can get temp1 & temp2 then i can solve it

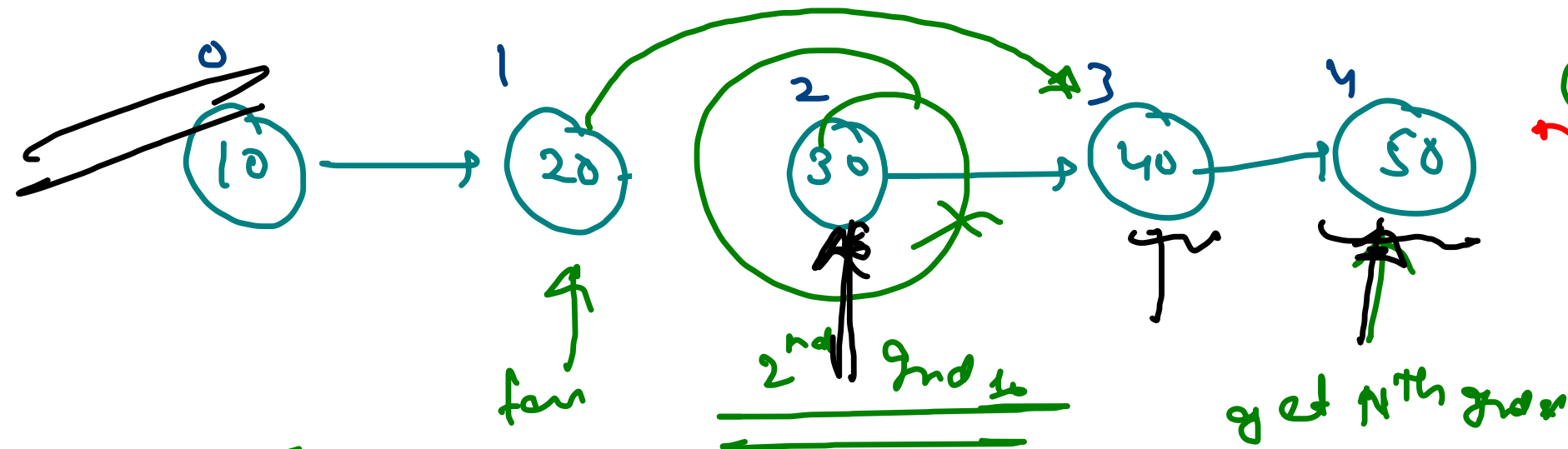
temp1 =  $(N-1)^{th}$  Node.

temp2 =  $N^{th}$  Node

find temp1 {  
 $temp1.next = nn;$   
 $nn.next = temp2;$

temp2 = temp1.next

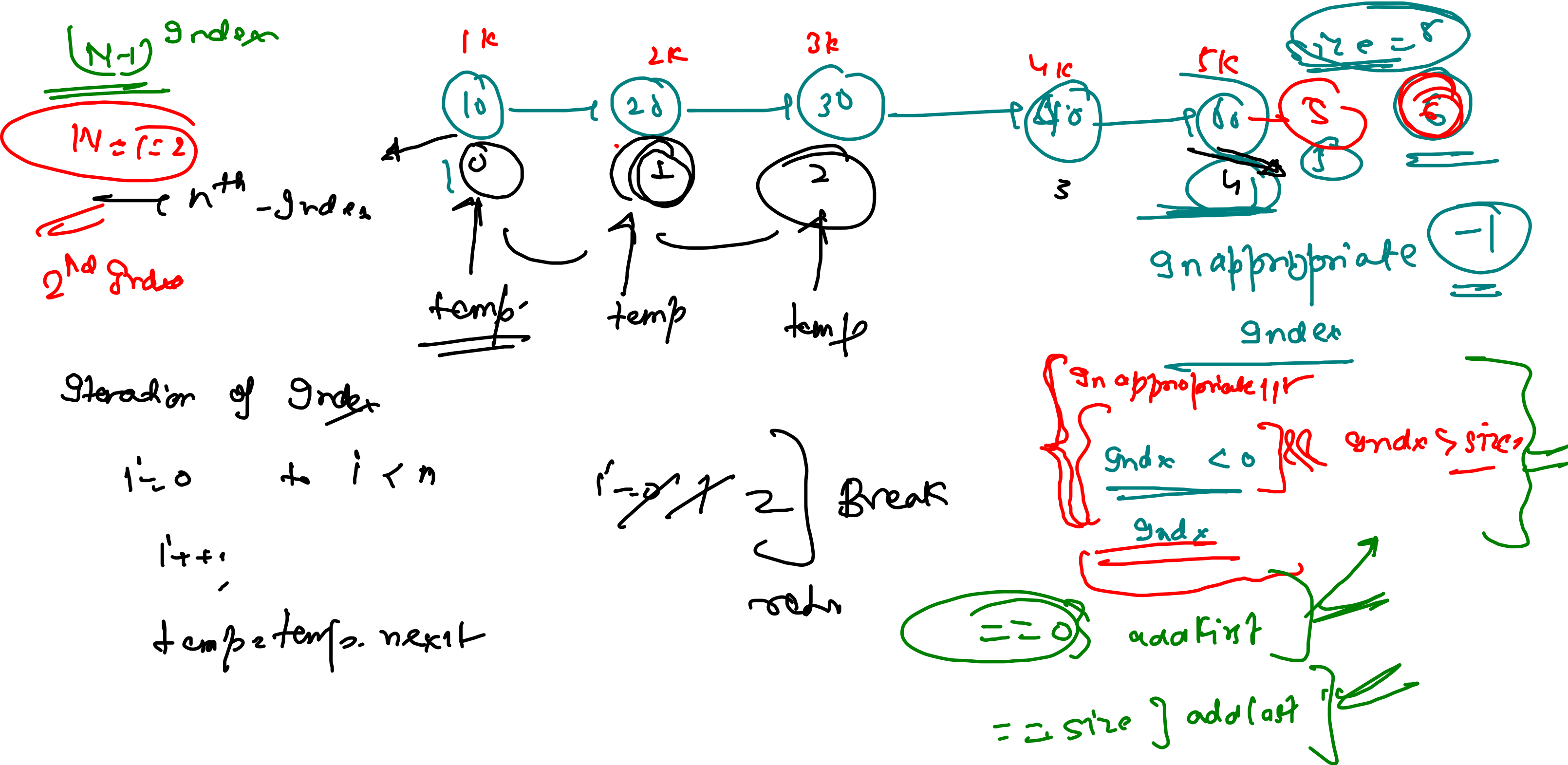
- Steps
- 1) find temp1
  - 2) make temp2
  - 3) make conn between



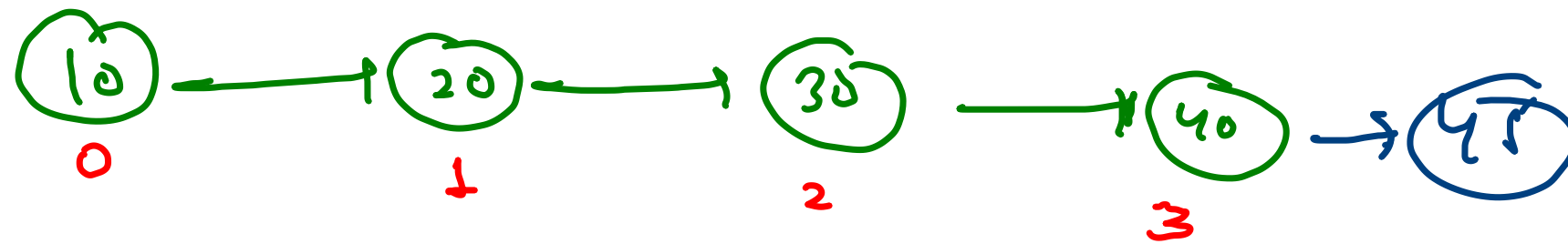
temp = 1st ind (n-1) ind

temp.next = temp.next.next

→ Inappropriate  
 → index  $\text{idx} < 0$   
 → Remove (index := 0)  
 → Remove first  
 → Remove (index := size - 1)  
 → Remove last ④  
 → Remove  $\in S$   
 0 2 4 6 8  
2







Size = 4

45 at 4