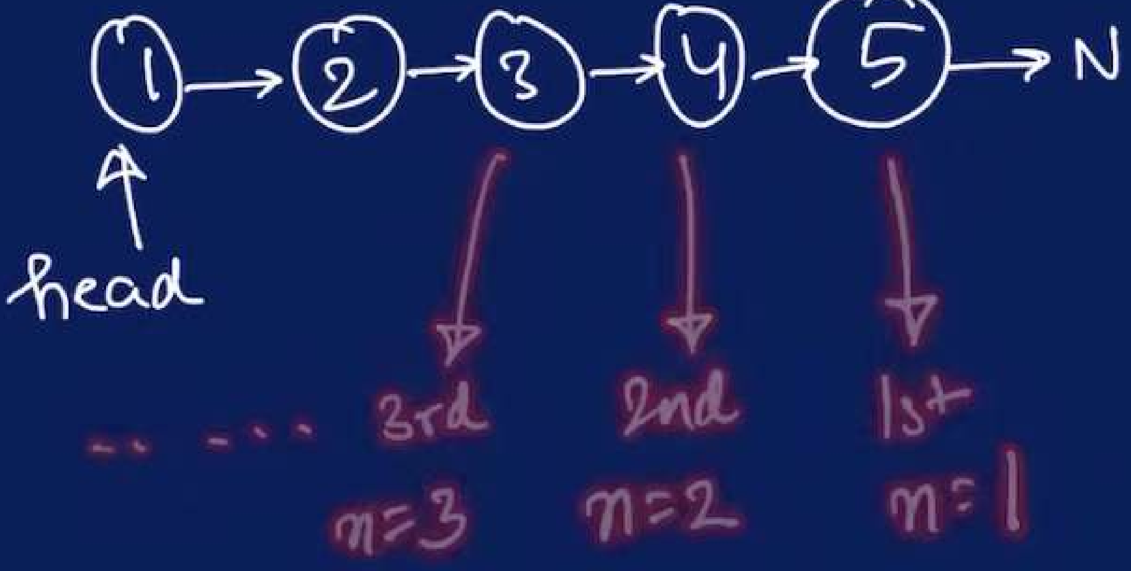
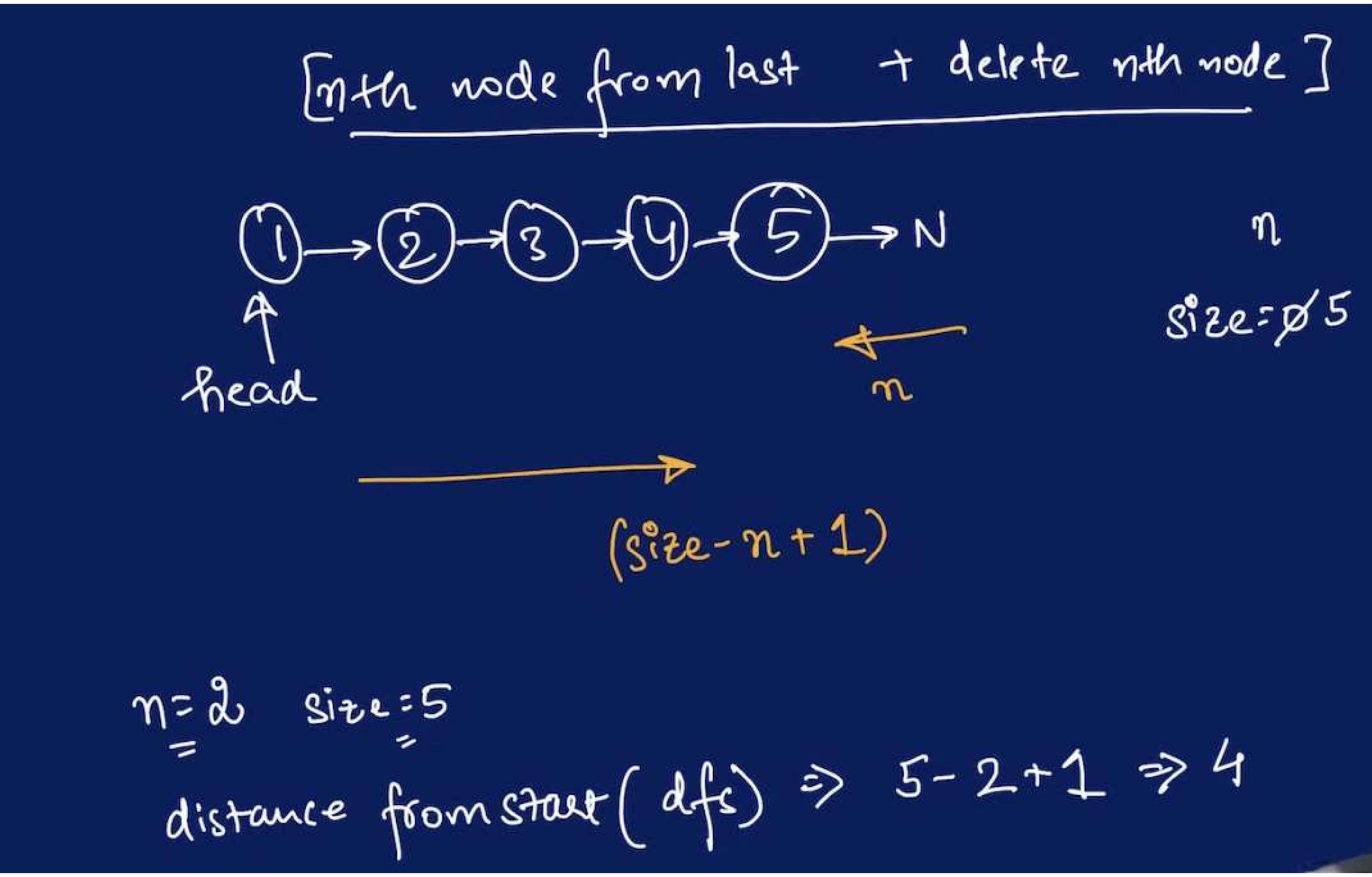
nth node from last





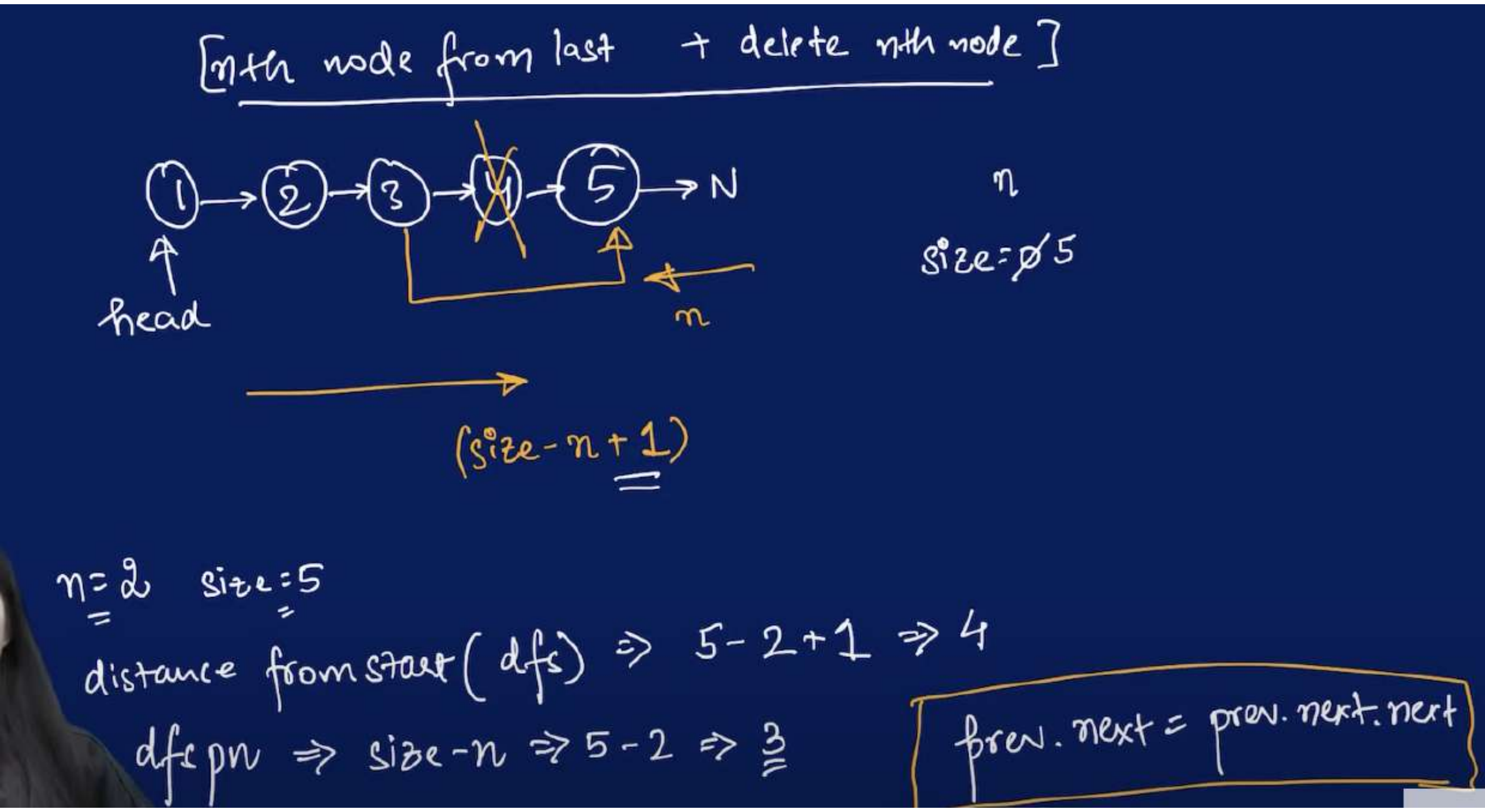
[nth node from last + delete nth node]

size = ~~0~~ 5

$(size - n + 1)$

$n = 2$ size $= 5$

distance from start (dfs) ⇒ $5 - 2 + 1$ ⇒ 4



[nth node from last + delete nth node]

n

size = ~~0~~ 5

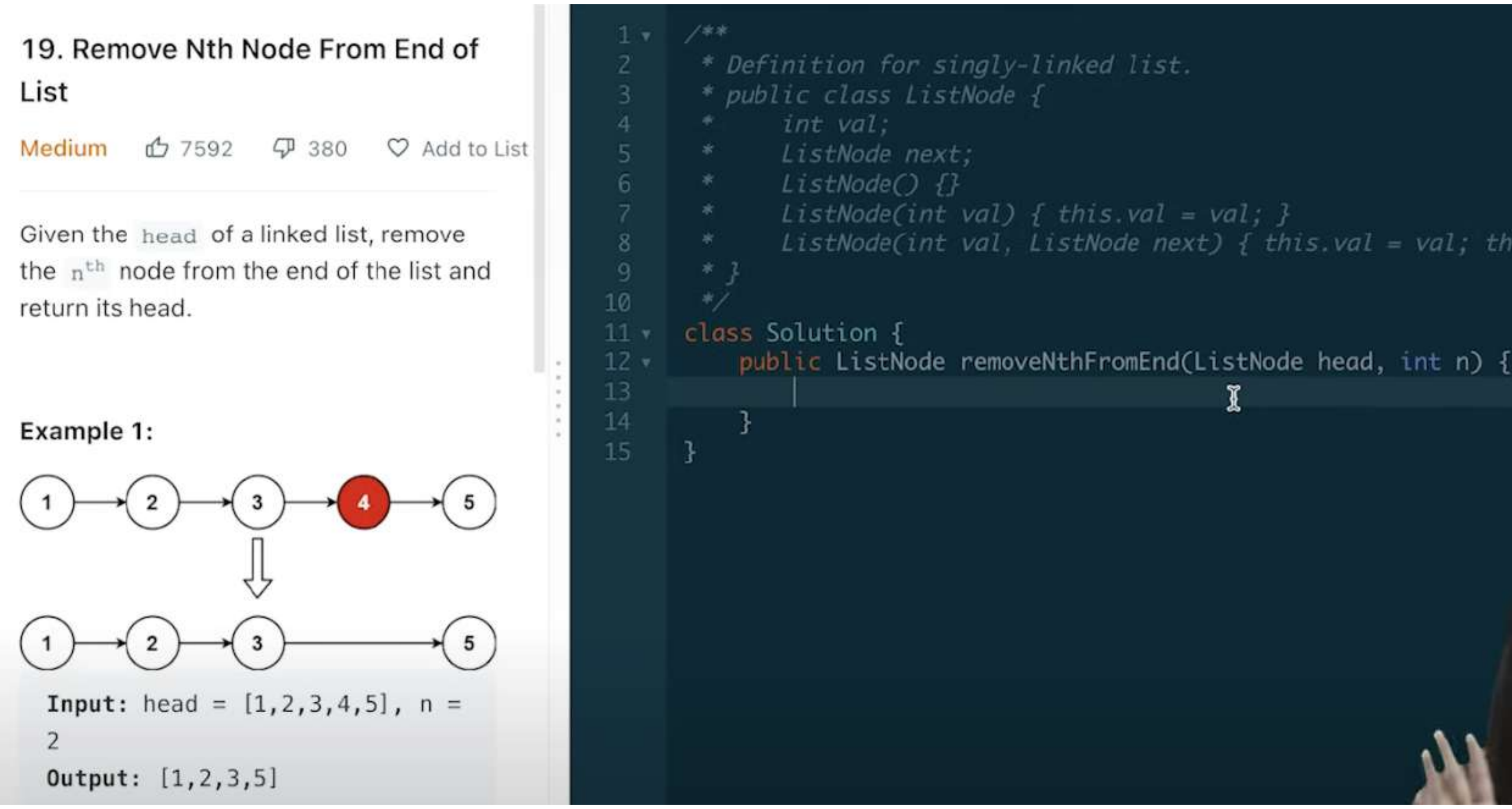
(size - n + 1)

n = 2   size = 5

distance from start (dfs) ⇒ 5 - 2 + 1 ⇒ 4

dfspn ⇒ size - n ⇒ 5 - 2 ⇒ 3

$\boxed{\text{prev.next = prev.next.next}}$



19. Remove Nth Node From End of List

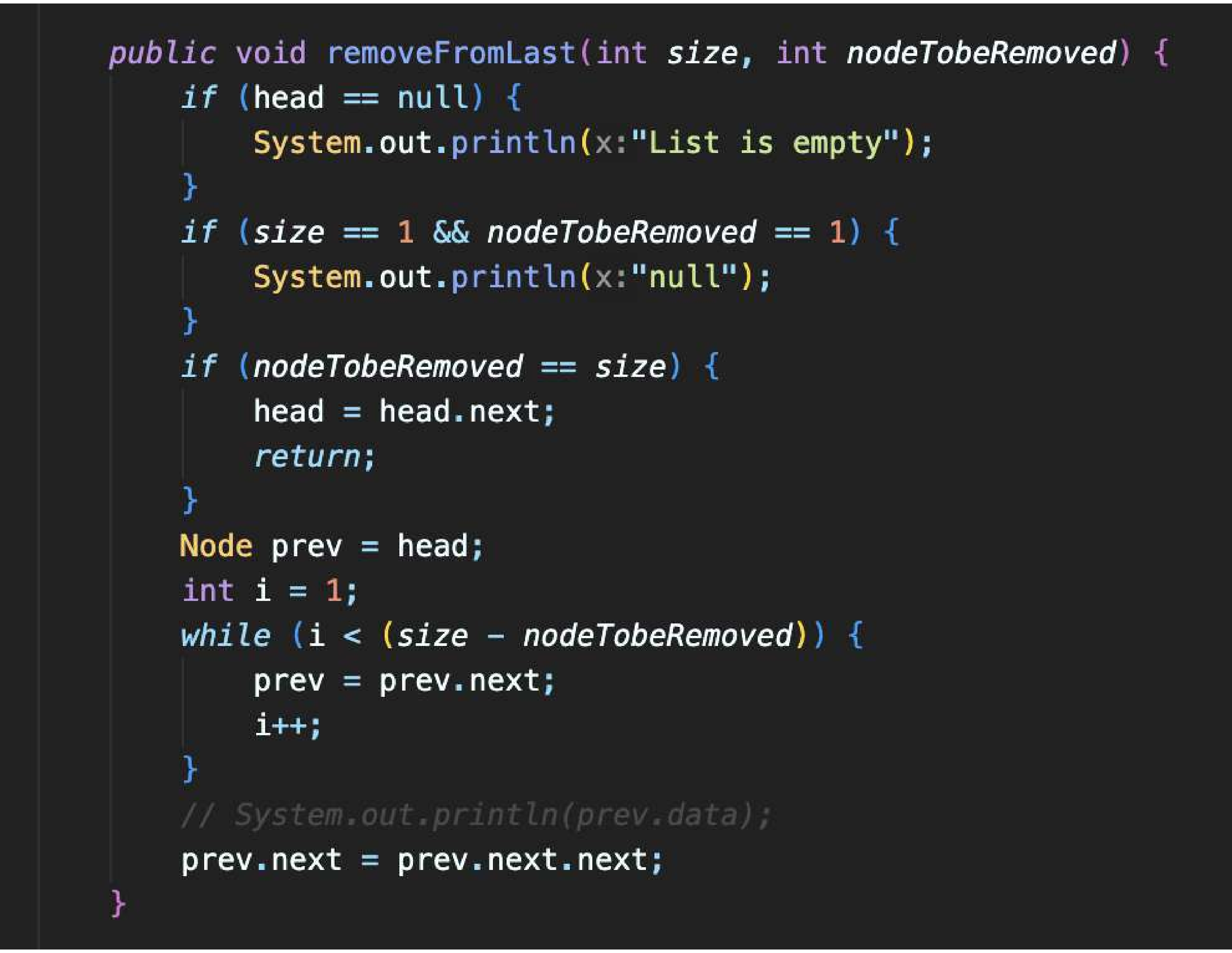
Given the `head` of a linked list, remove the n^{th} node from the end of the list and return its head.

Example 1:

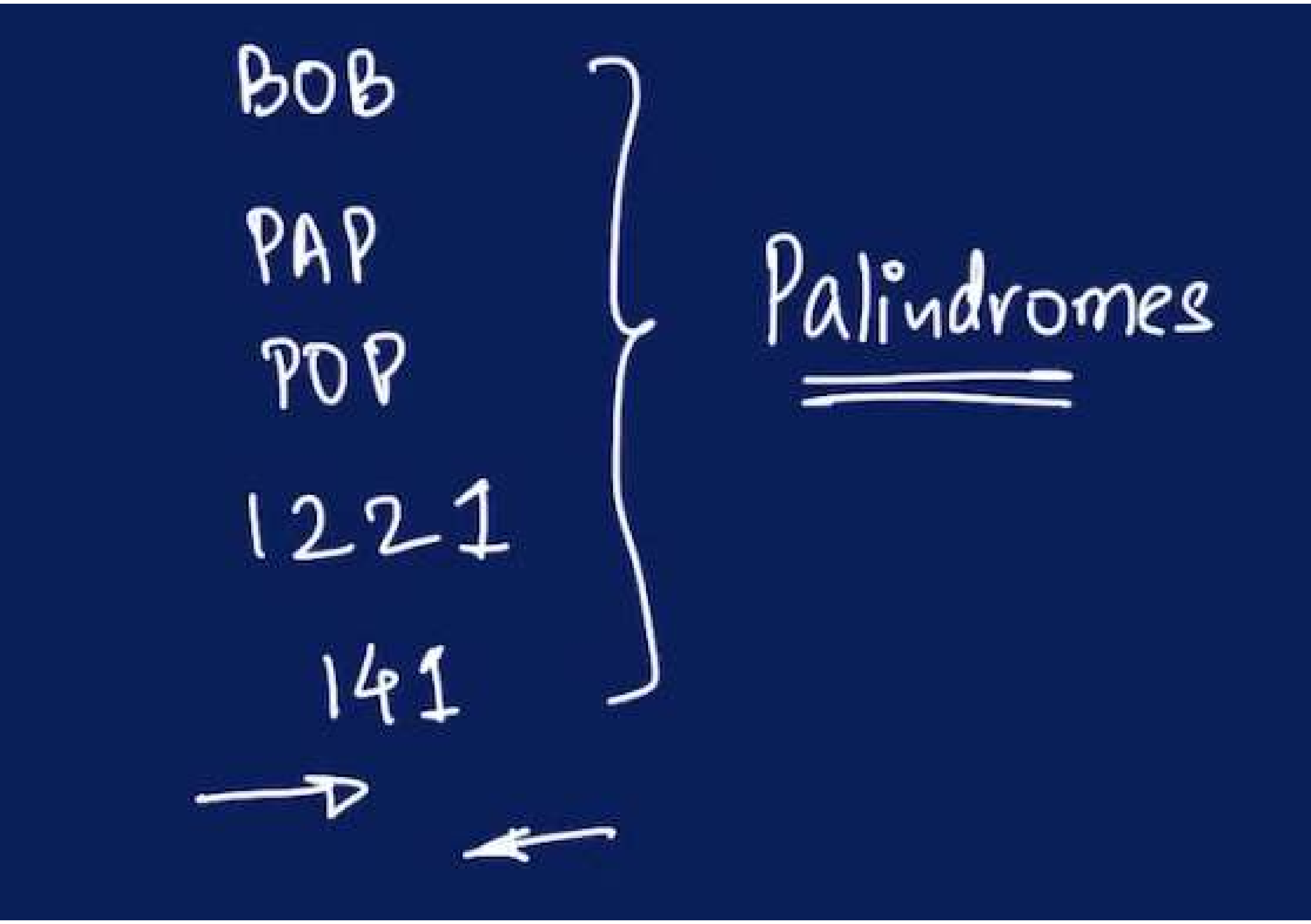
```
Input: head = [1,2,3,4,5], n =
2
Output: [1,2,3,5]
```

```java
/**
 * Definition for singly-linked list.
 * public class ListNode {
 *     int val;
 *     ListNode next;
 *     ListNode() {}
 *     ListNode(int val) { this.val = val; }
 *     ListNode(int val, ListNode next) { this.val = val; th
 * }
 */
class Solution {
    public ListNode removeNthFromEnd(ListNode head, int n) {

    }
}
```



```java
public void removeFromLast(int size, int nodeTobeRemoved) {
    if (head == null) {
        System.out.println(x:"List is empty");
    }
    if (size == 1 && nodeTobeRemoved == 1) {
        System.out.println(x:"null");
    }
    if (nodeTobeRemoved == size) {
        head = head.next;
        return;
    }
    Node prev = head;
    int i = 1;
    while (i < (size - nodeTobeRemoved)) {
        prev = prev.next;
        i++;
    }
    // System.out.println(prev.data);
    prev.next = prev.next.next;
}
```



BOB

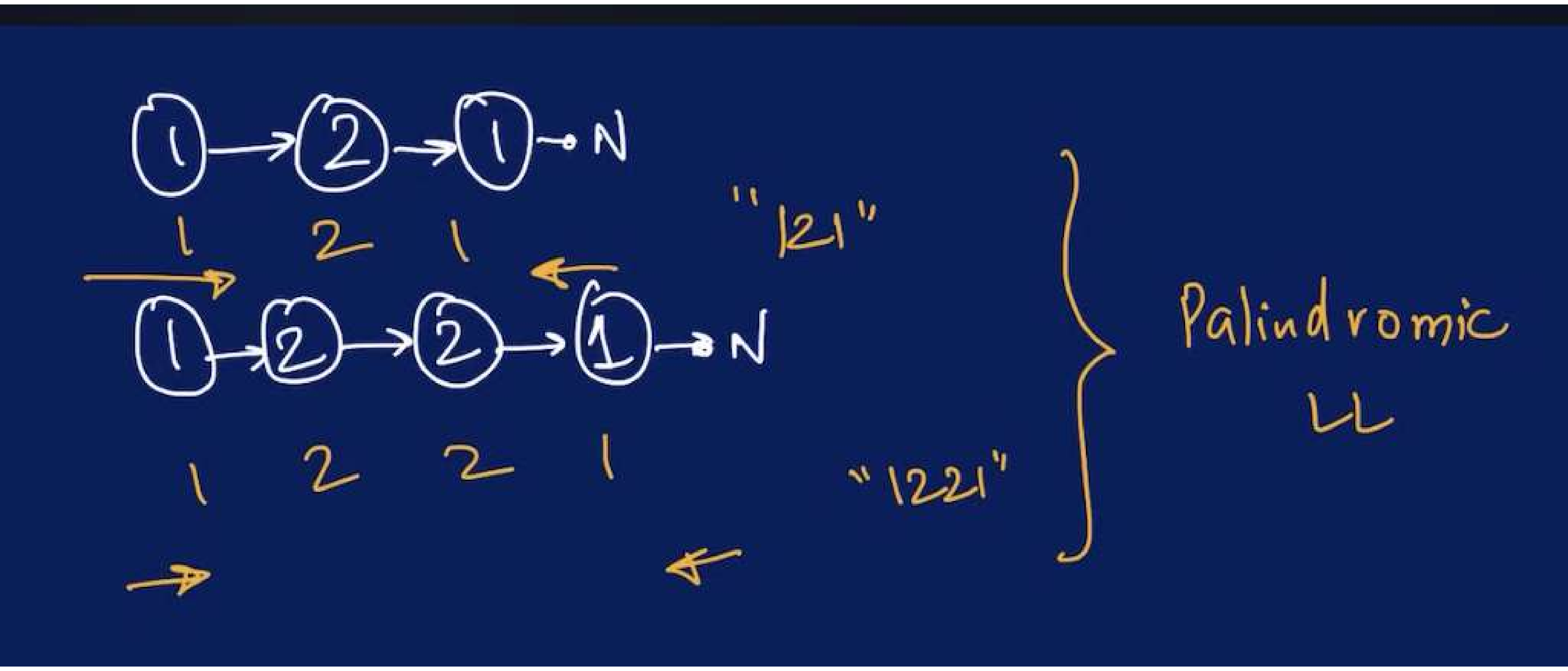
PAP

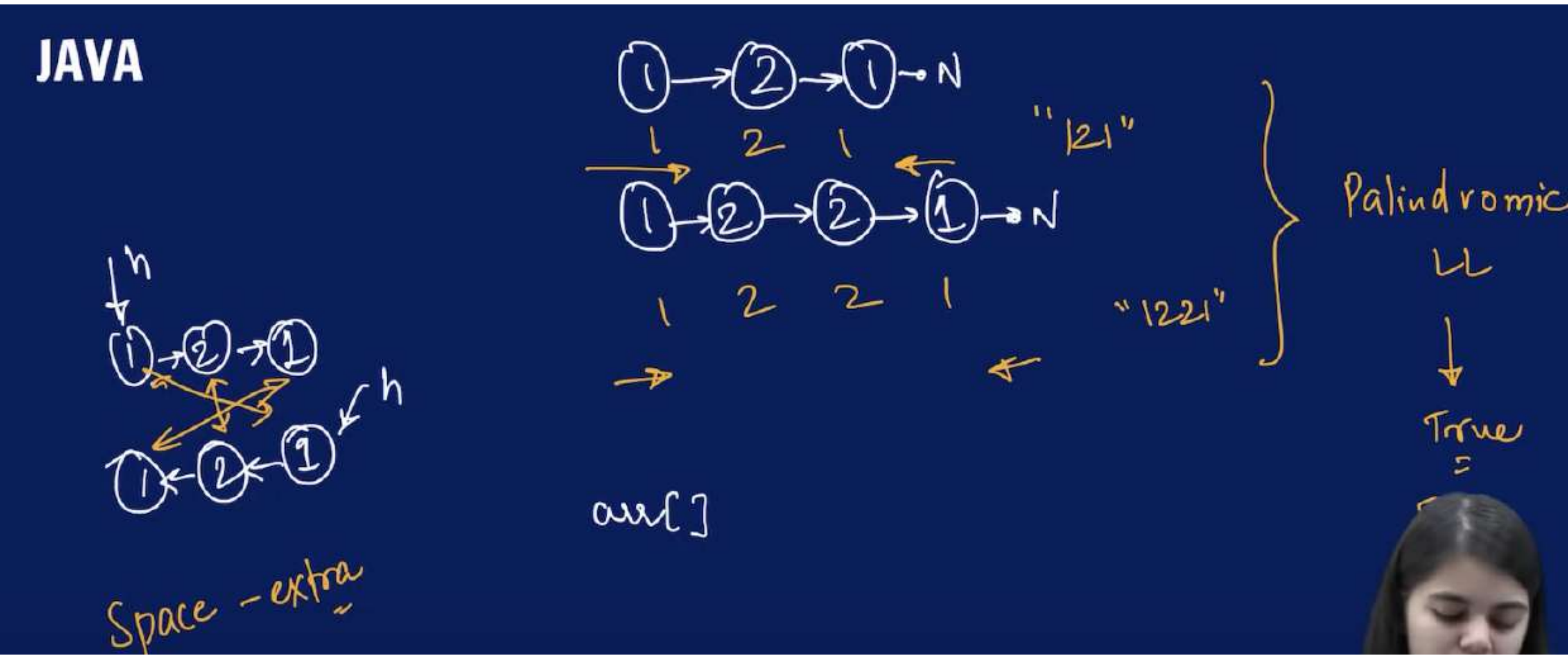
POP

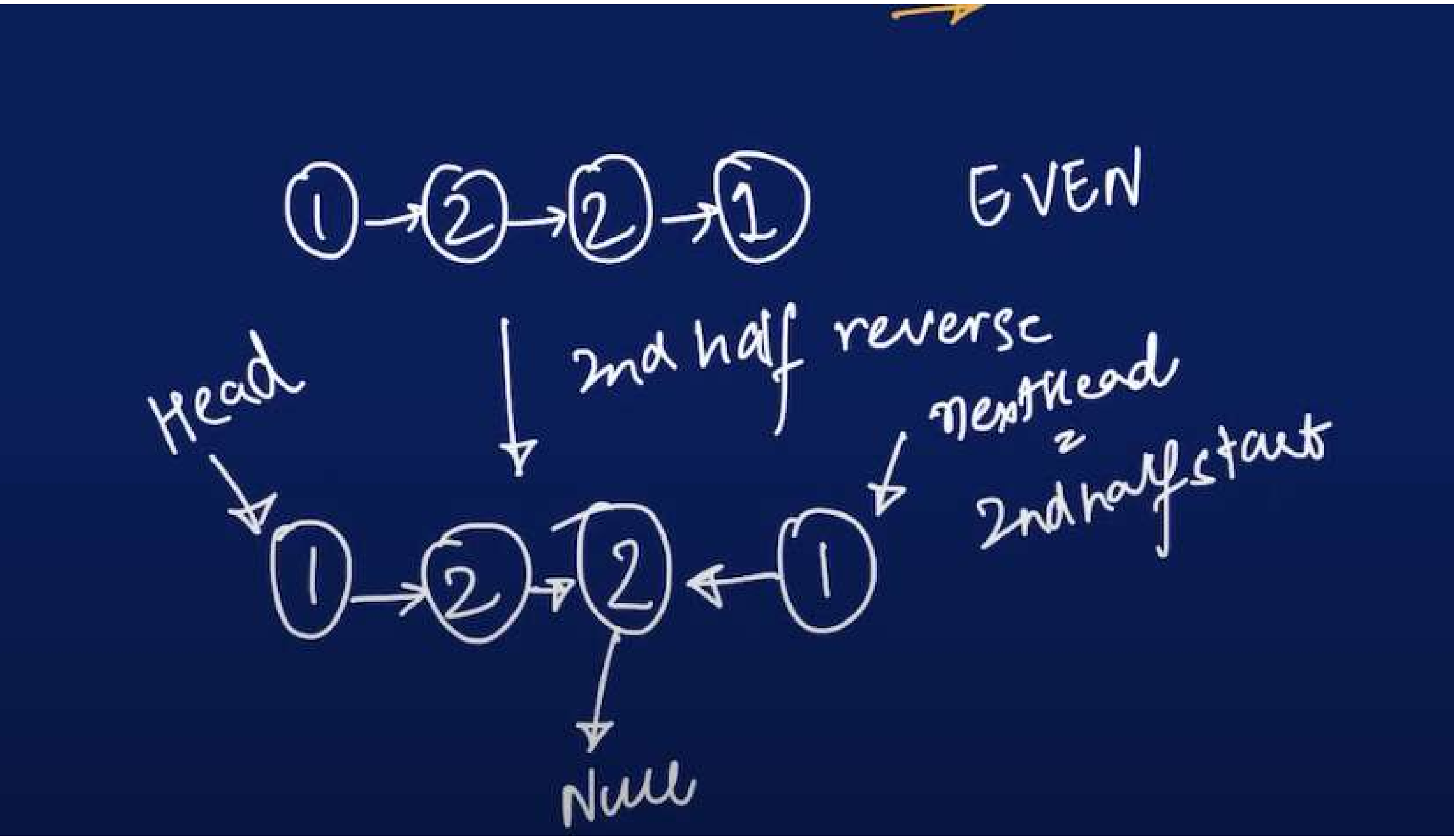
1221

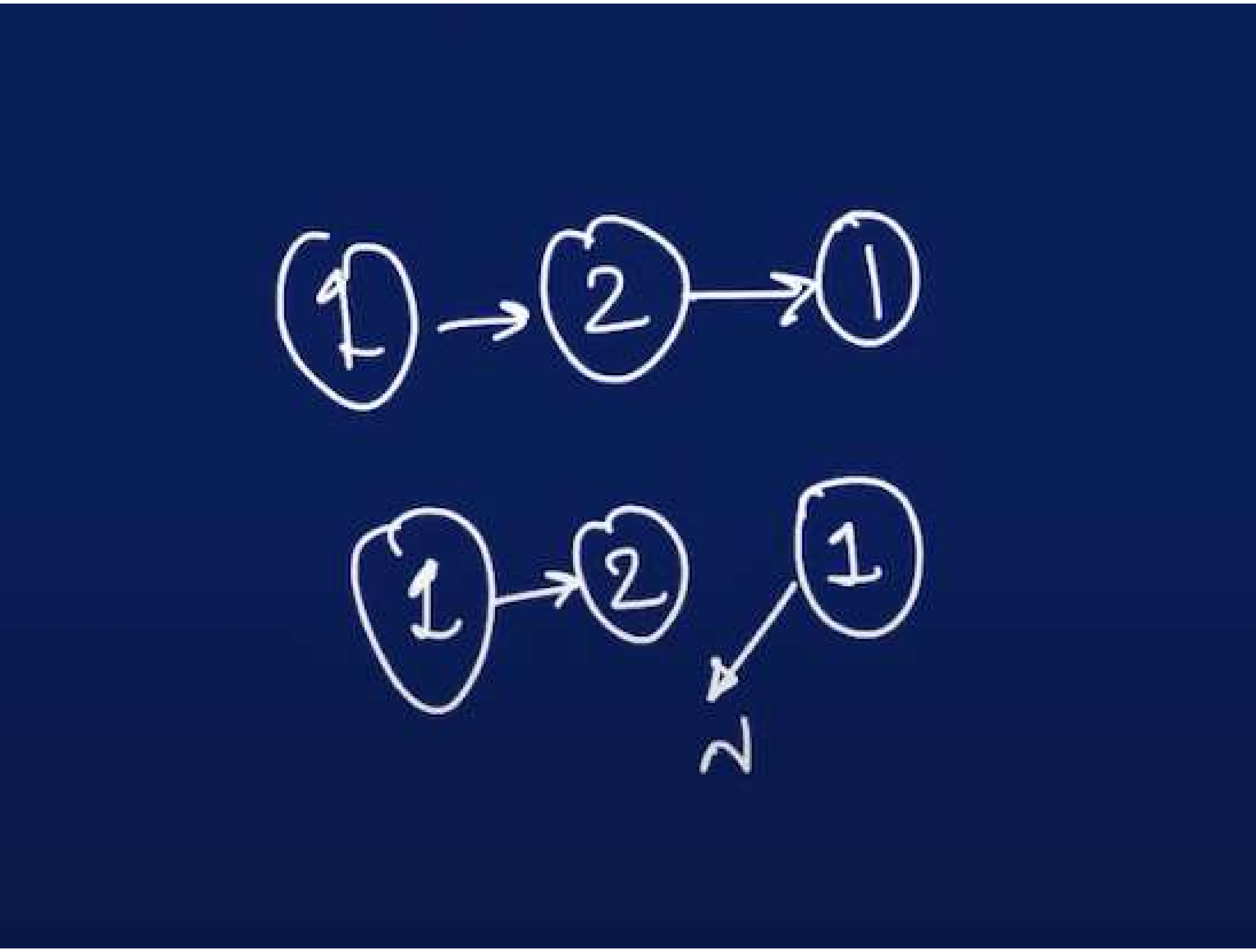
141

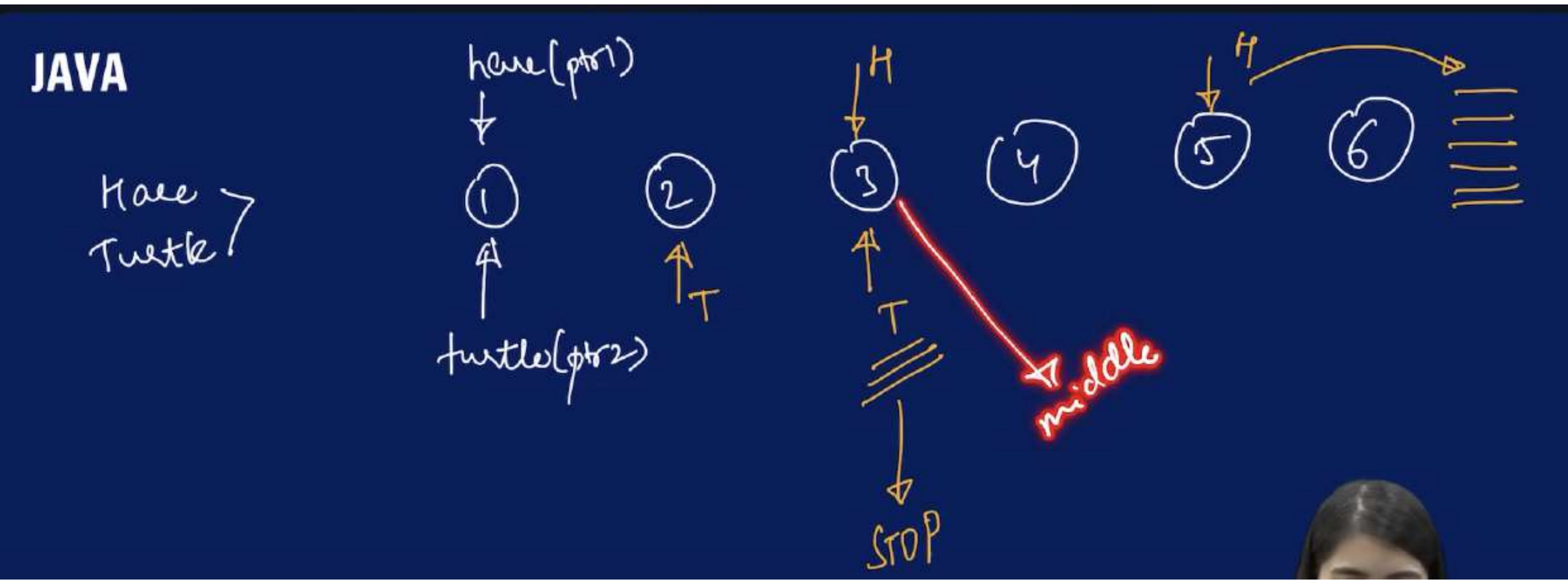
Palindromes

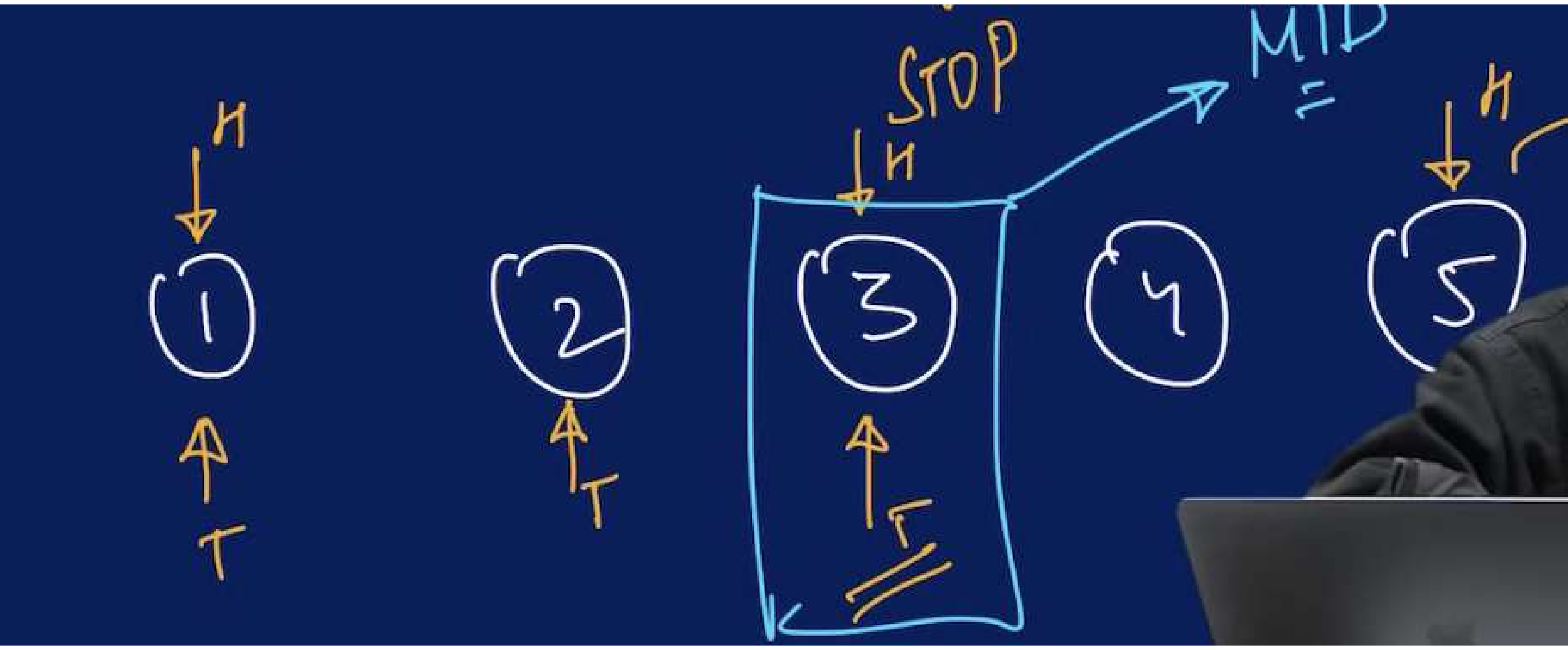

1 → 2 → 1 → N
1 2 1
"121"
1 → 2 → 2 → 1 → N
1 2 2 1
"1221"
Palindromic LL

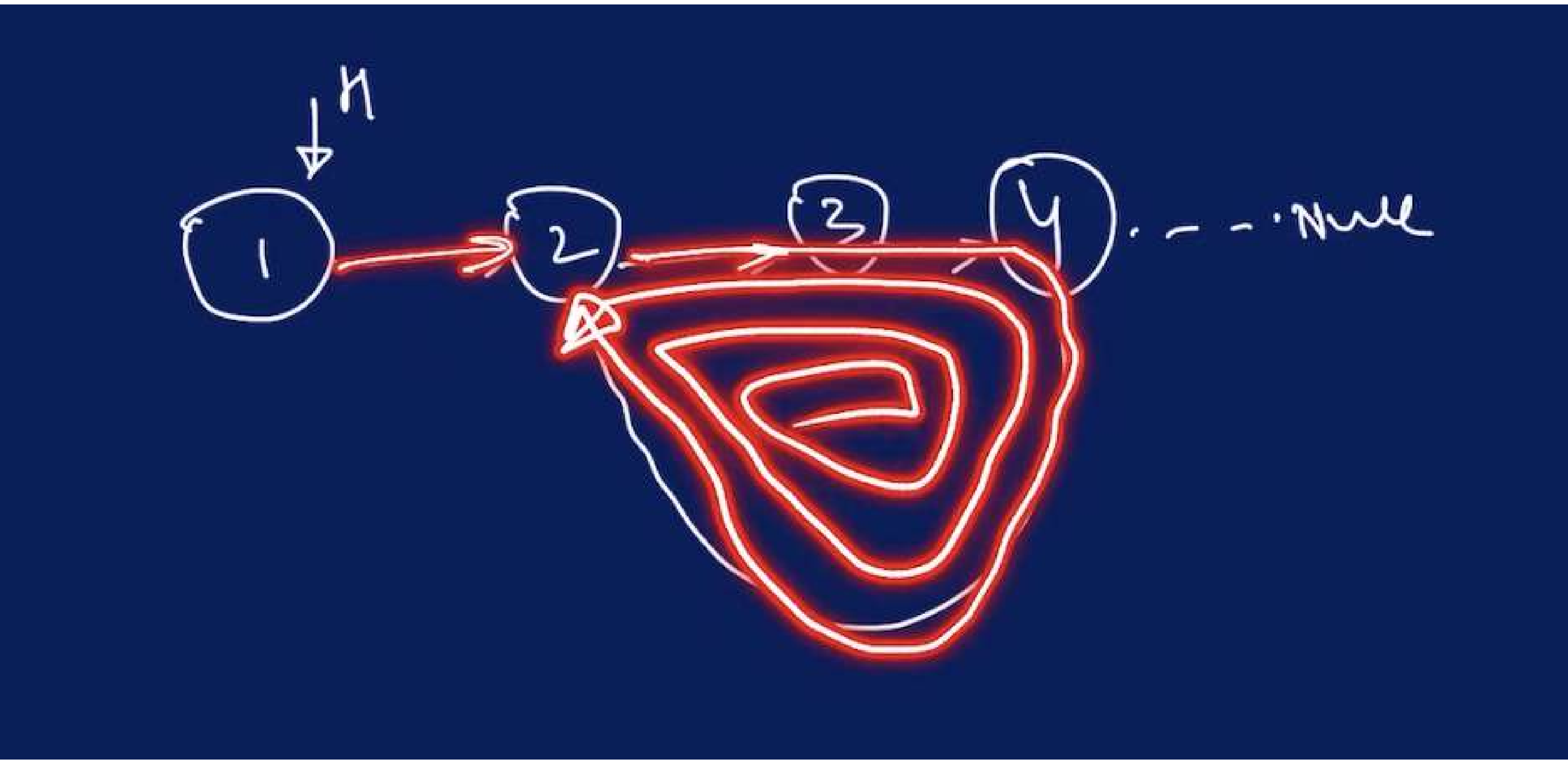

JAVA
1 → 2 → 1 → N
1 2 1
"121"
1 → 2 → 2 → 1 → N
1 2 2 1
"1221"
Palindromic LL
True
h
1 → 2 → 1
1 ← 2 ← 1
h
arr[ ]
Space - extra

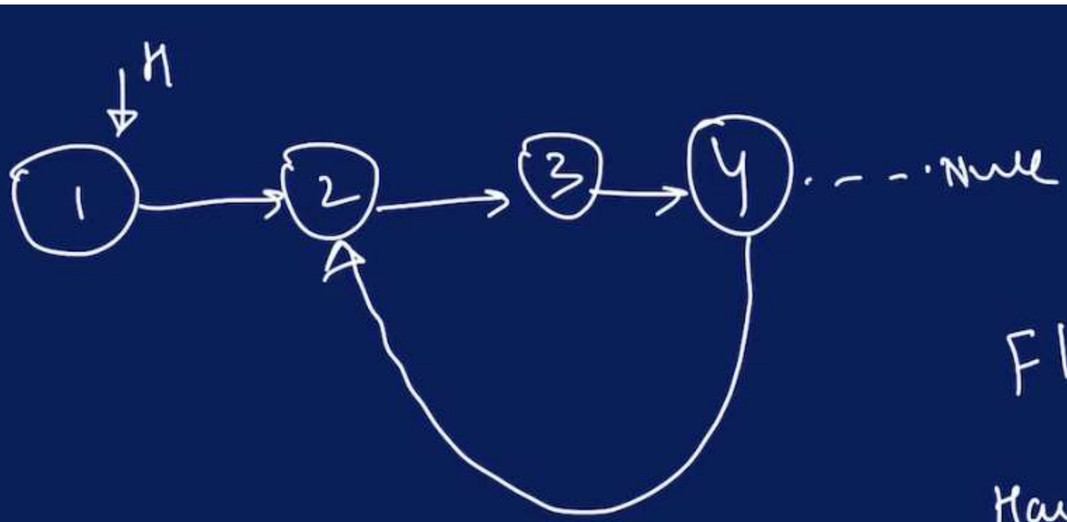

1
2
2
1
EVEN
2nd half reverse
Head
nextHead
=
2nd half start
1
2
2
1
Null


1
2
1
1
2
1
2


JAVA
Hare
Turtle
hare(ptr1)
H
H
1
2
3
4
5
6
T
T
turtle(ptr2)
middle
STOP

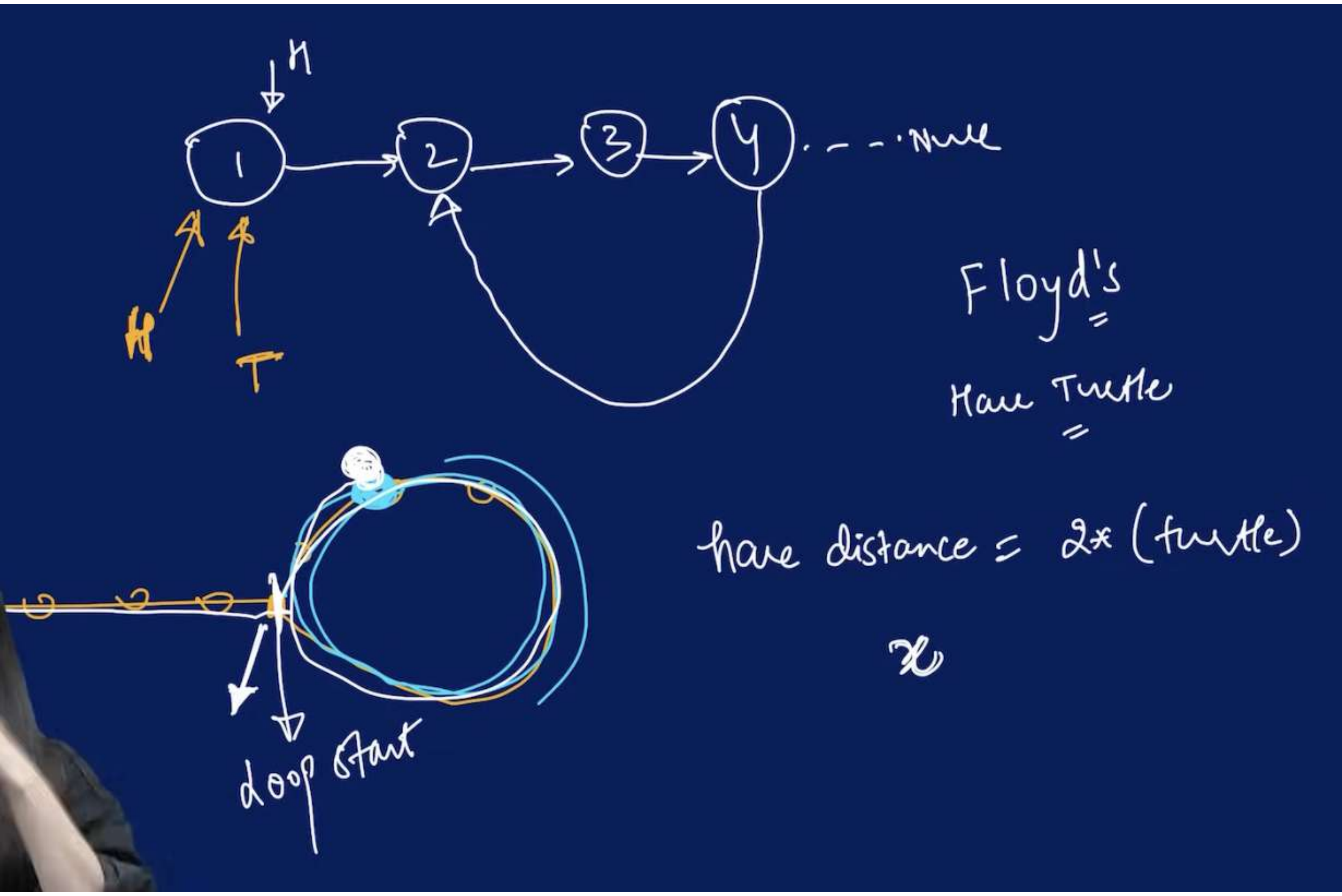

STOP
MID
=
H
H
H
1
2
3
4
5
T
T
T


H
1
2
3
4
Null



Floyd's

Hare Turtle

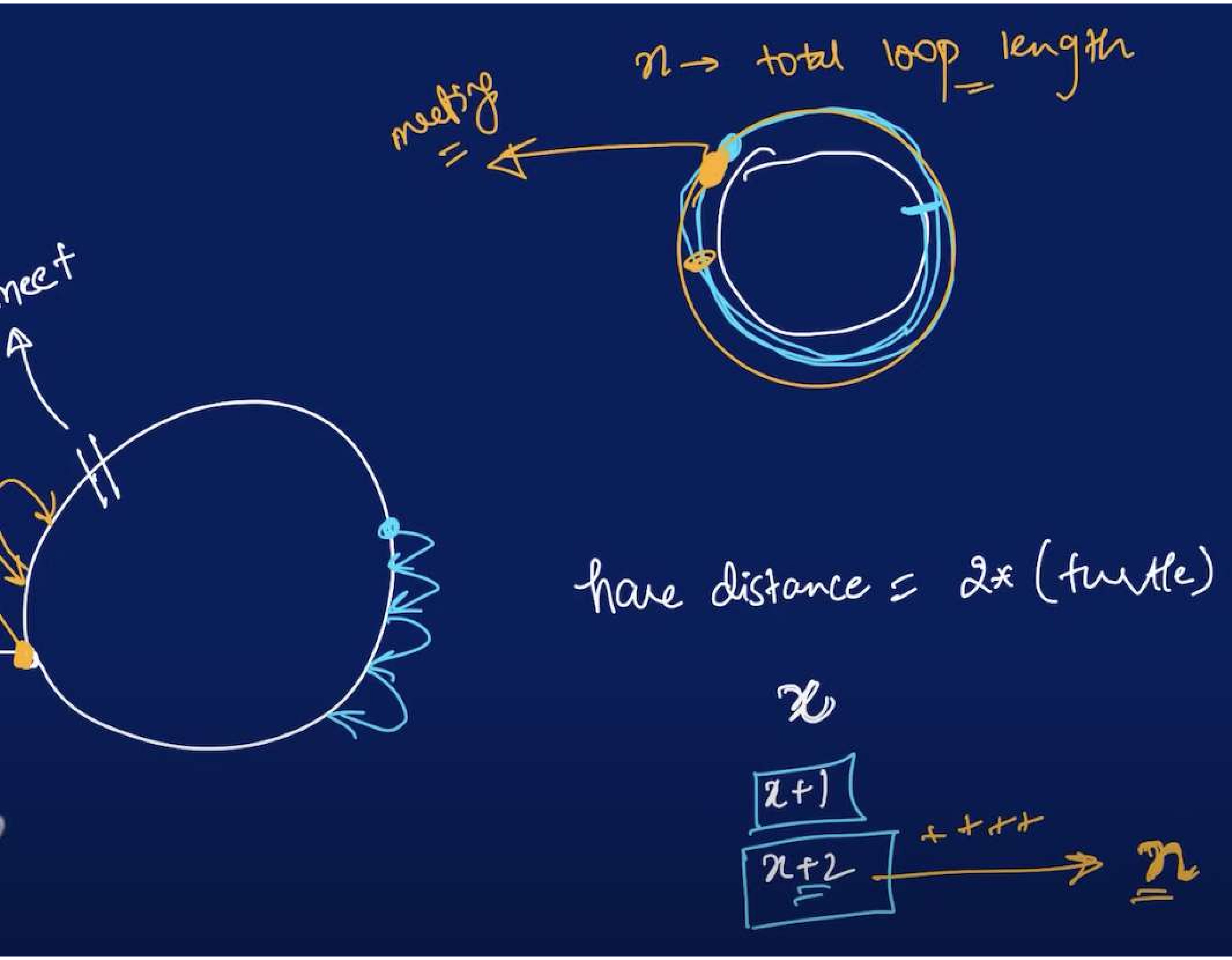


Floyd's

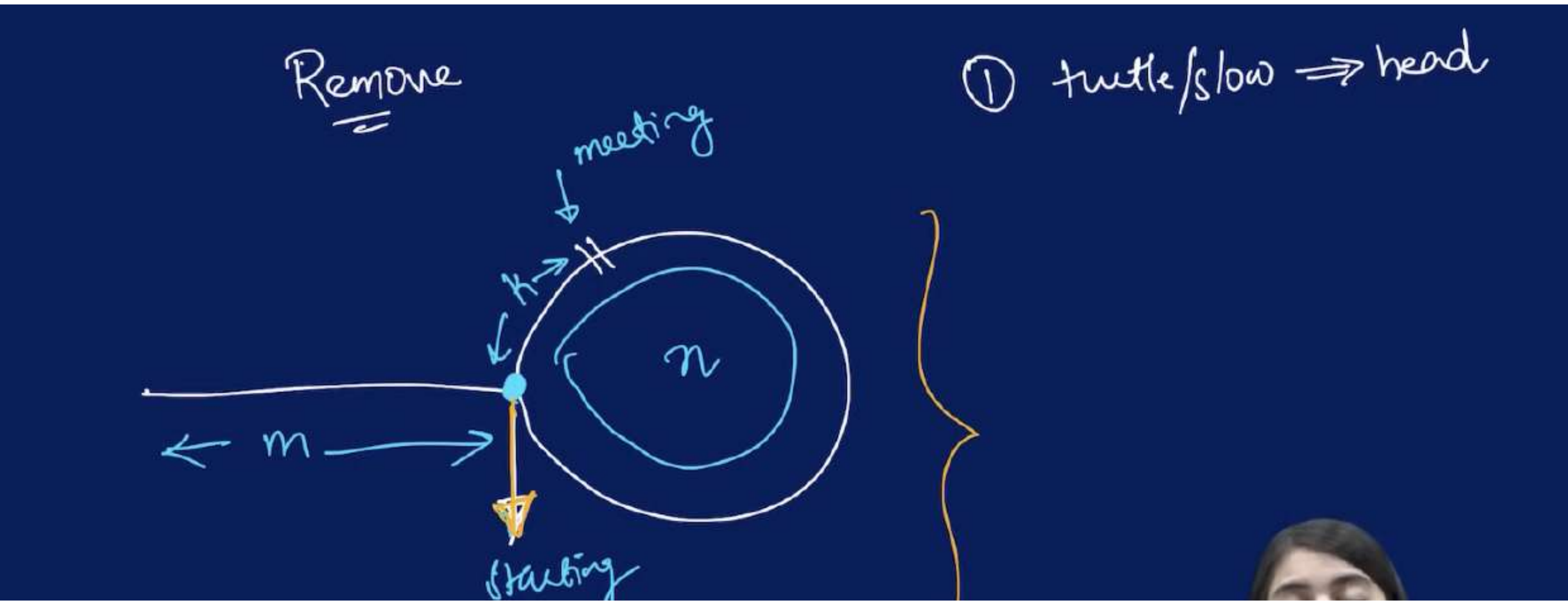
Hare Turtle

hare distance = 2* (turtle)

x



have distance = 2* (turtle)



Remove

① turtle/slow ⇒ head