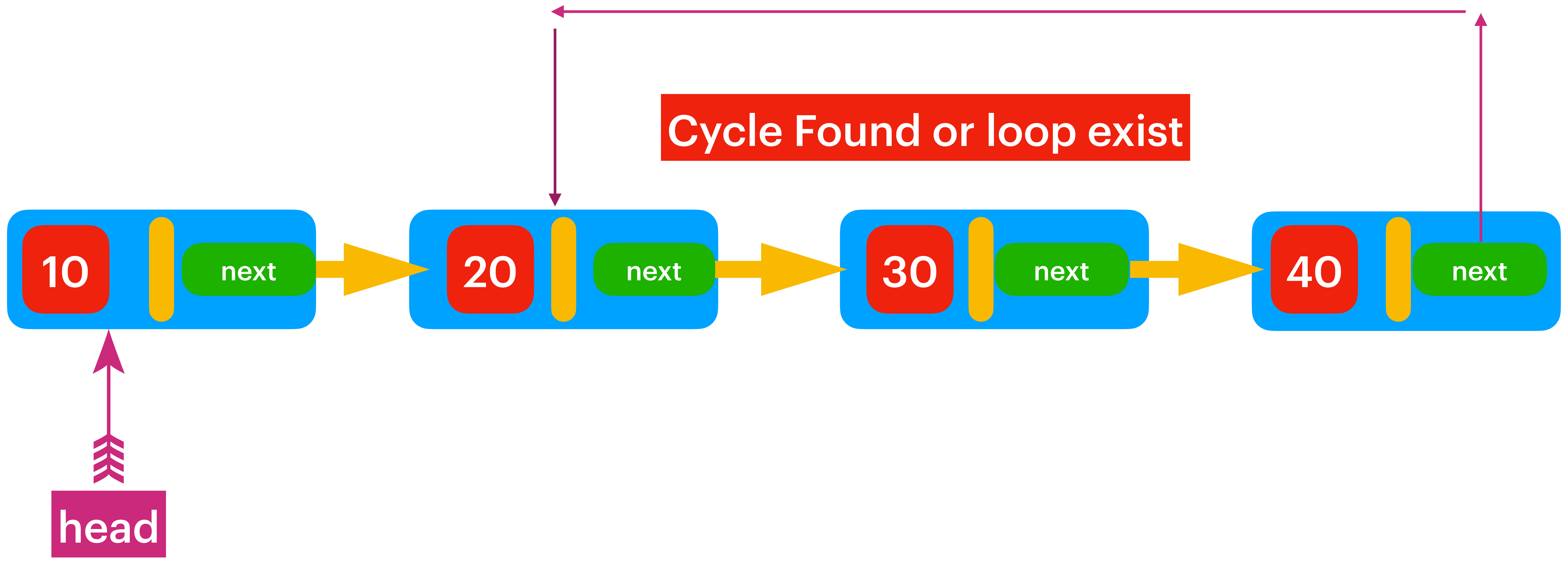
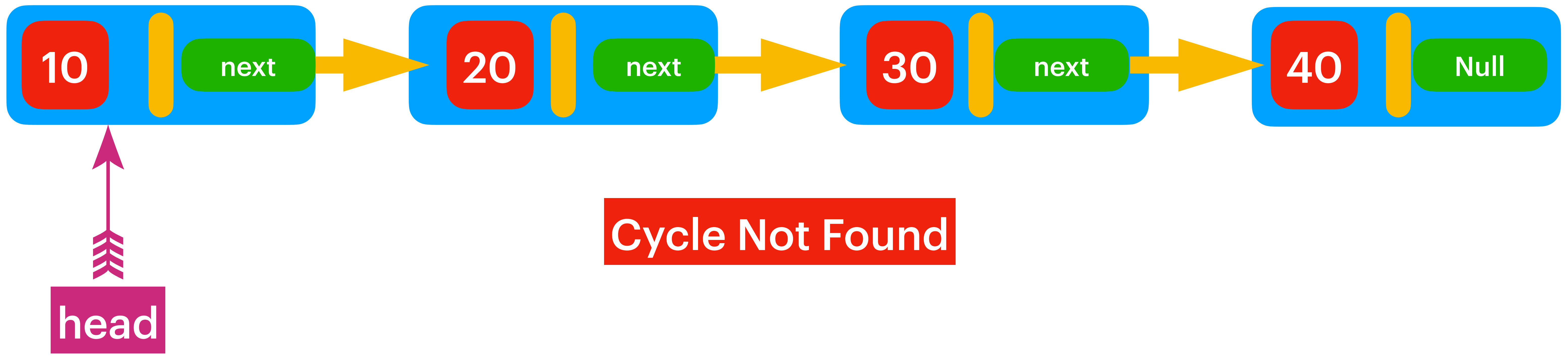
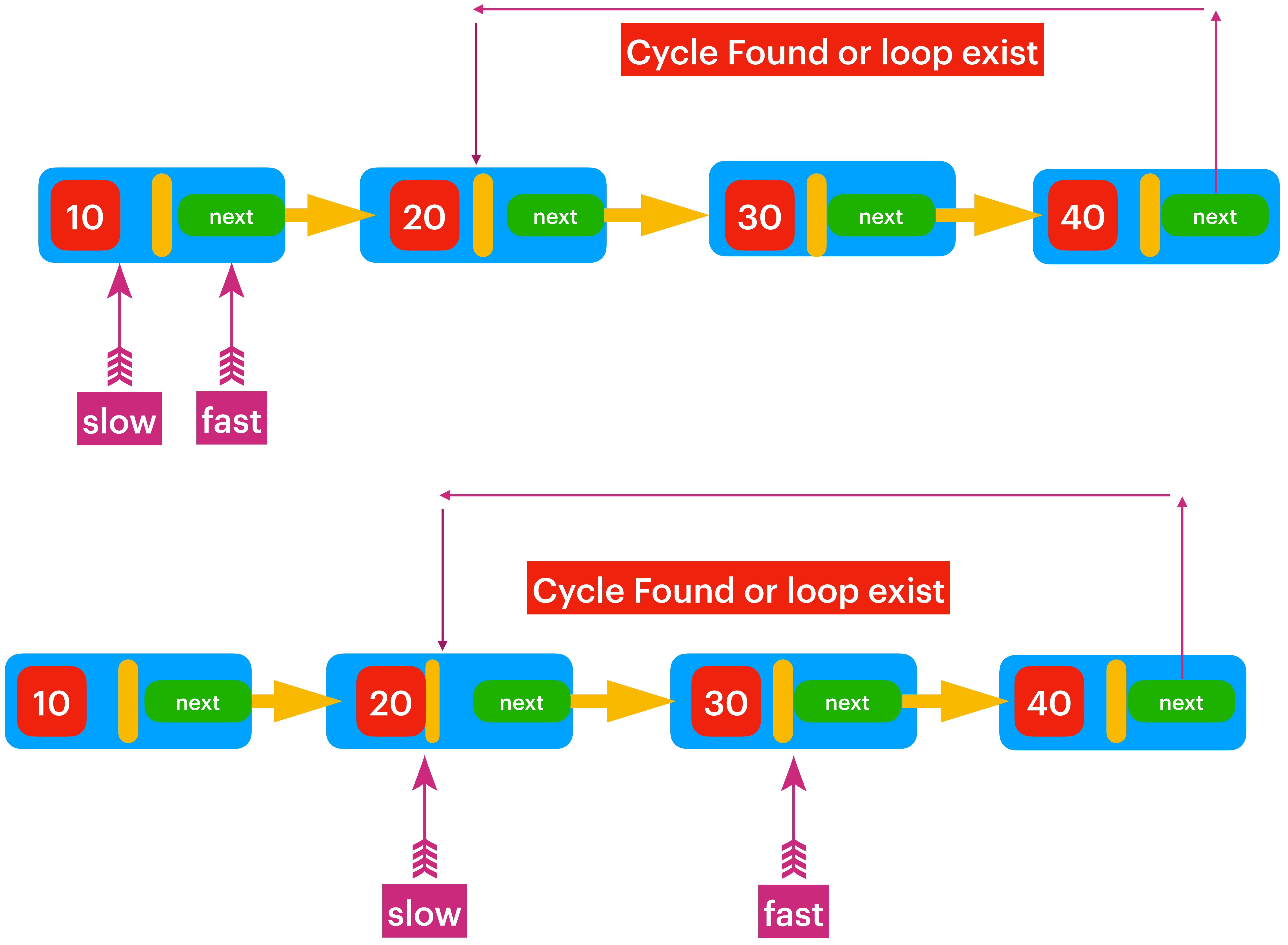
Detect the Cycle In List





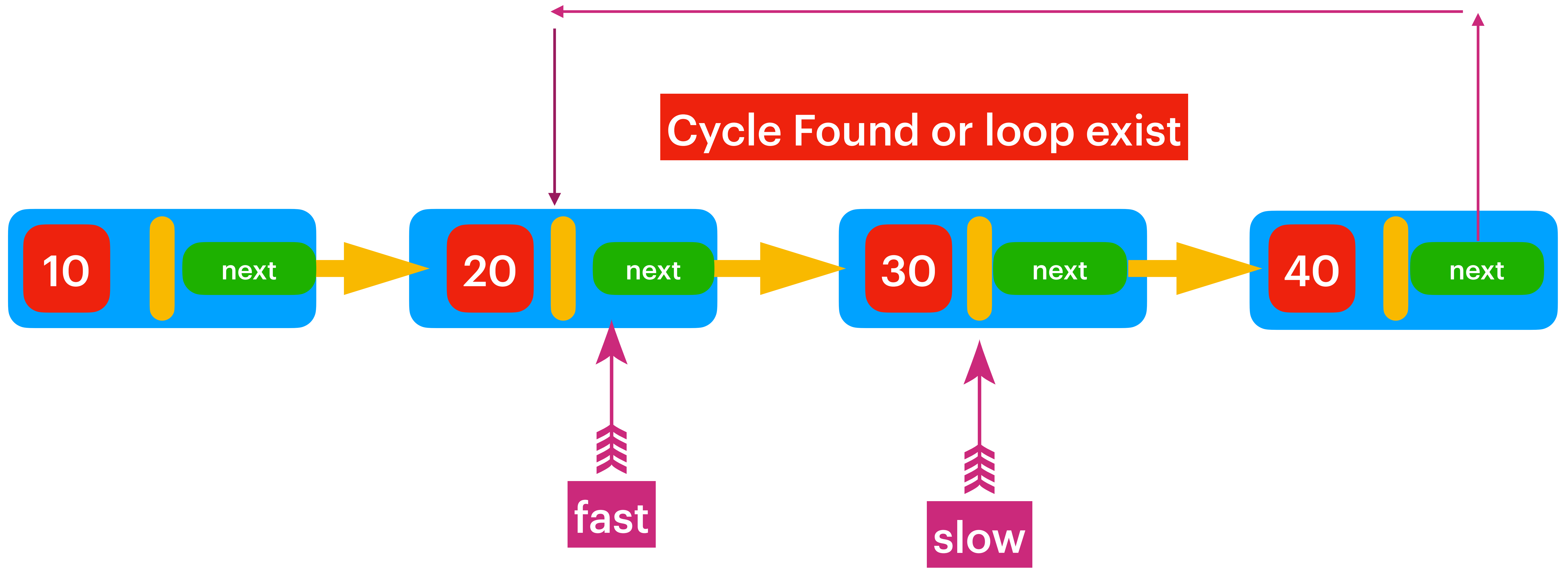
Detect the Cycle In List

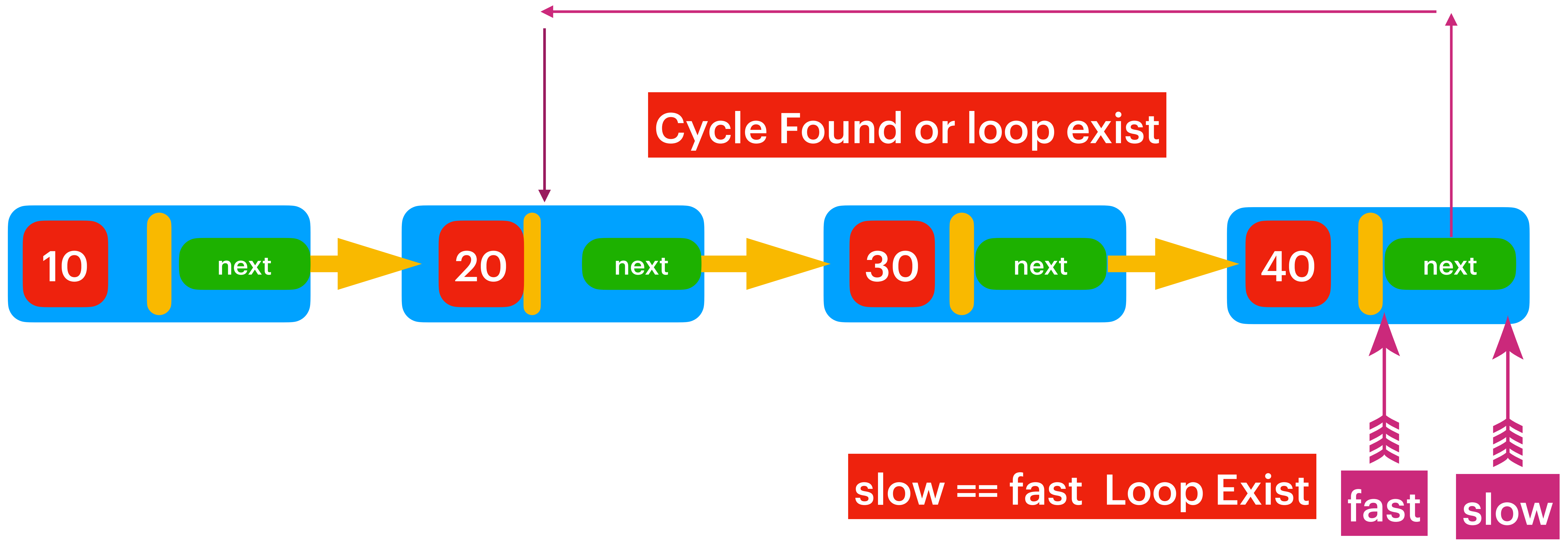
slow pointer jumps 1 step at a time.
fast pointer jumps 2 steps at a time.



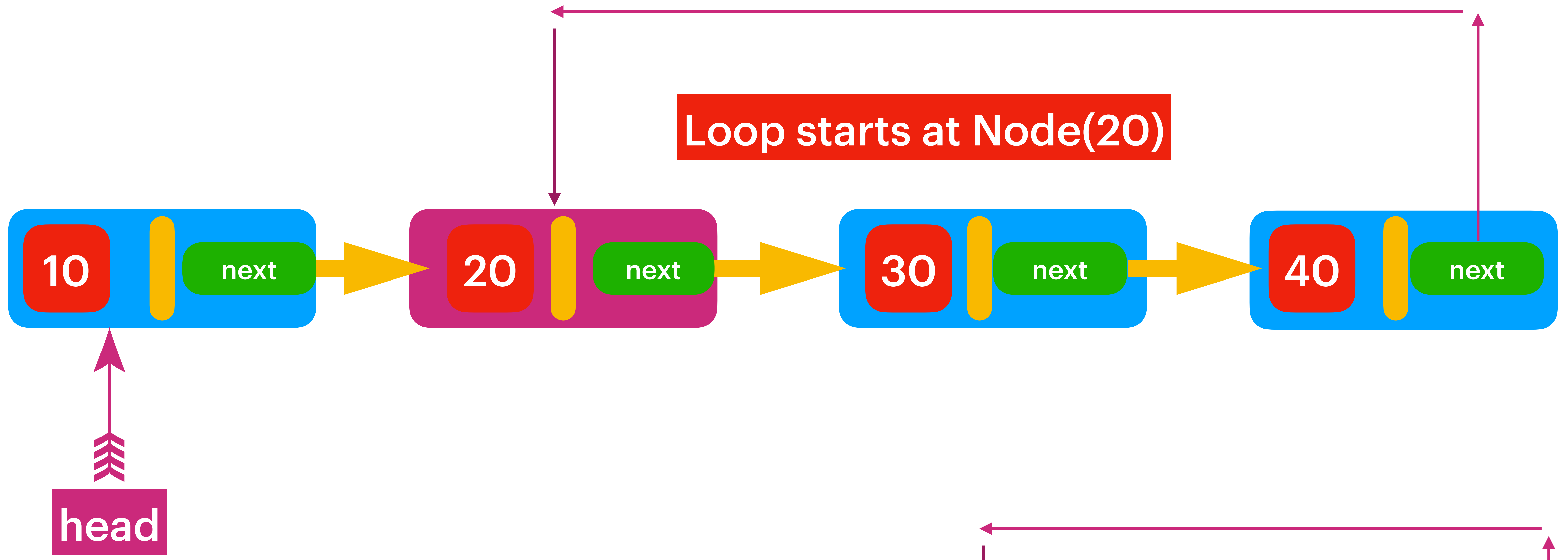
Detect the Cycle In List

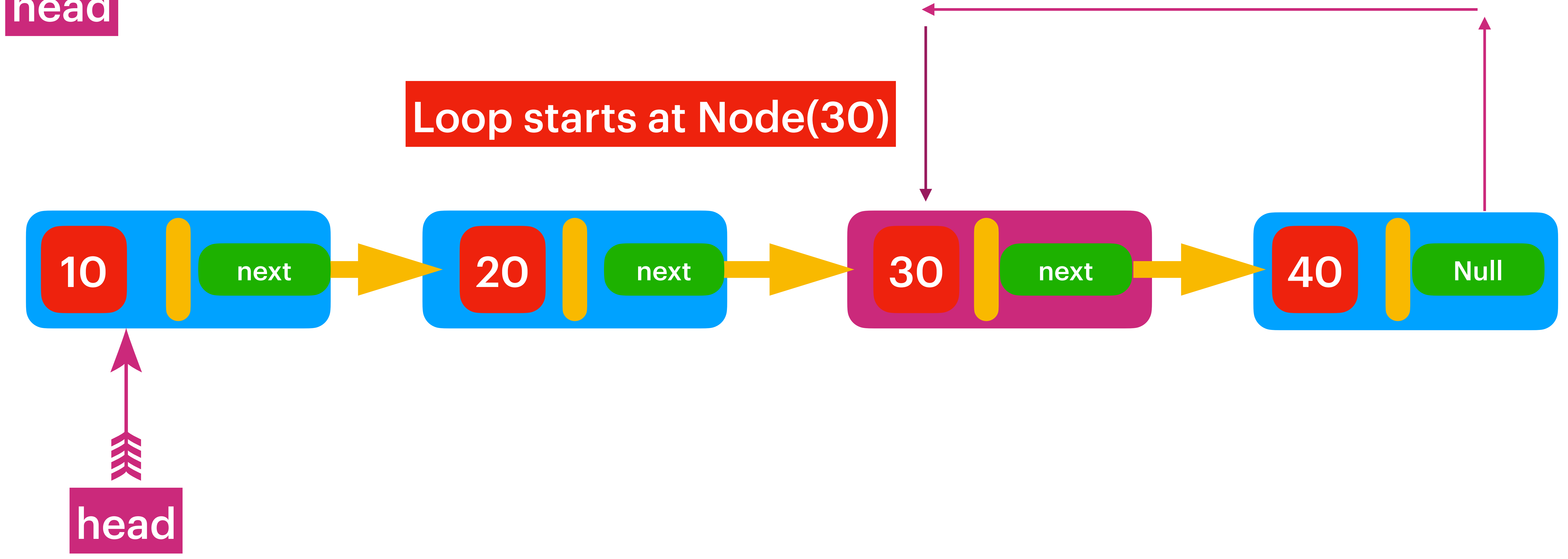
slow pointer jumps 1 step.
fast pointer jumps 2 steps at a time.





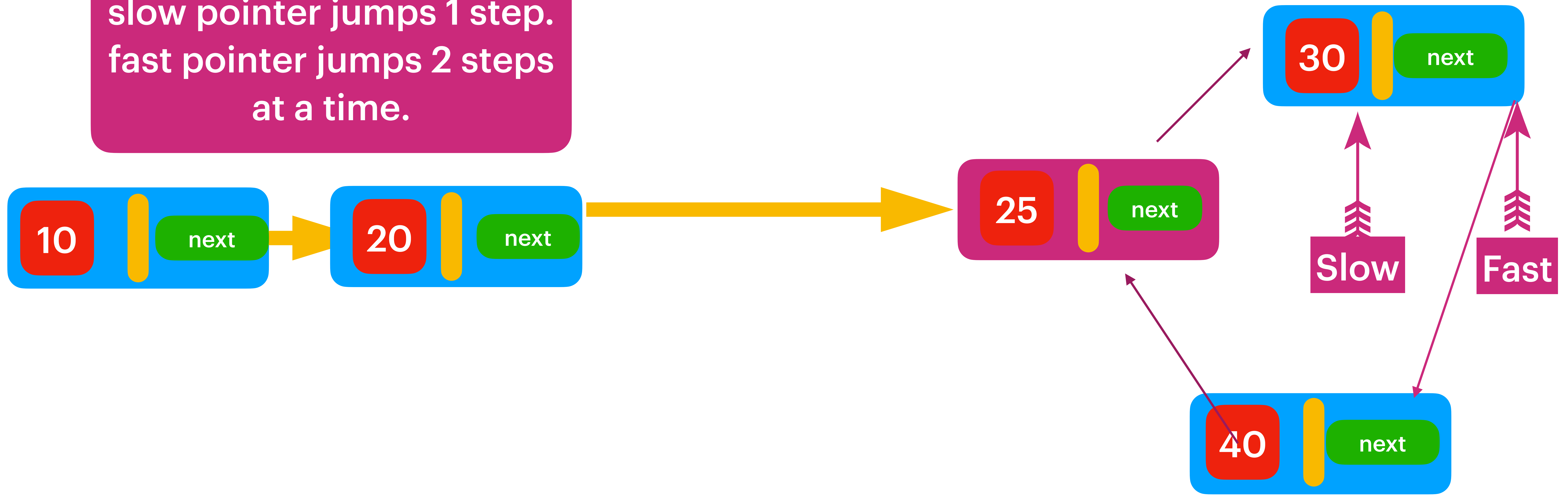
FindOut Loop Starting point



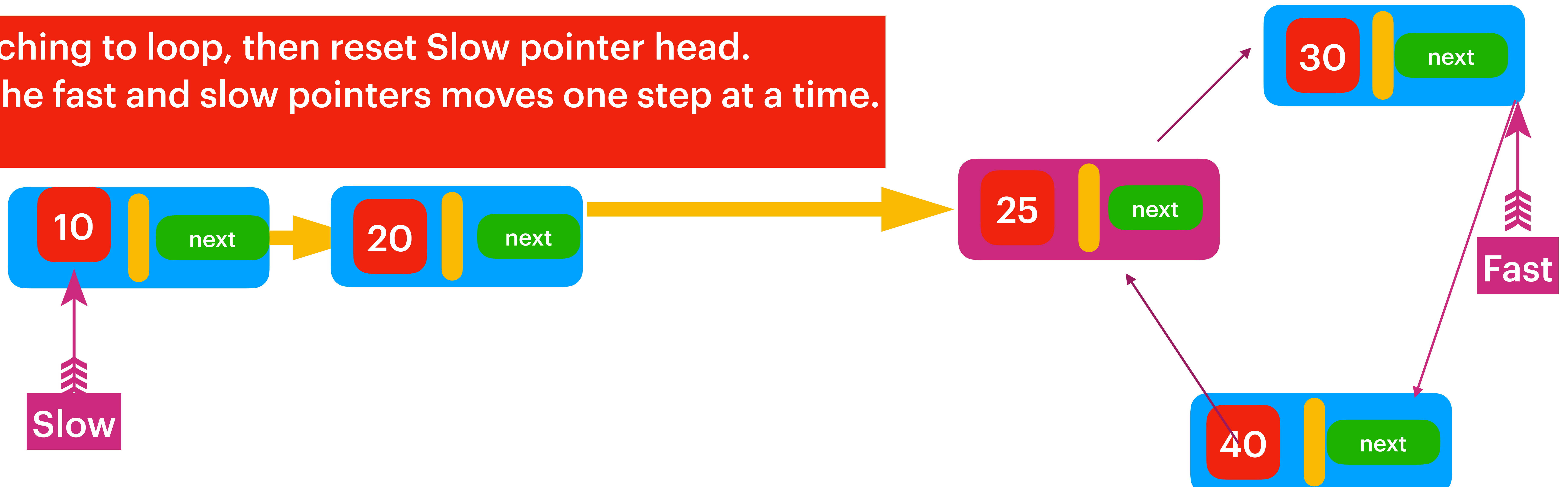


FindOut Loop Starting point

First find out the meeting point in a loop.
slow pointer jumps 1 step.
fast pointer jumps 2 steps at a time.

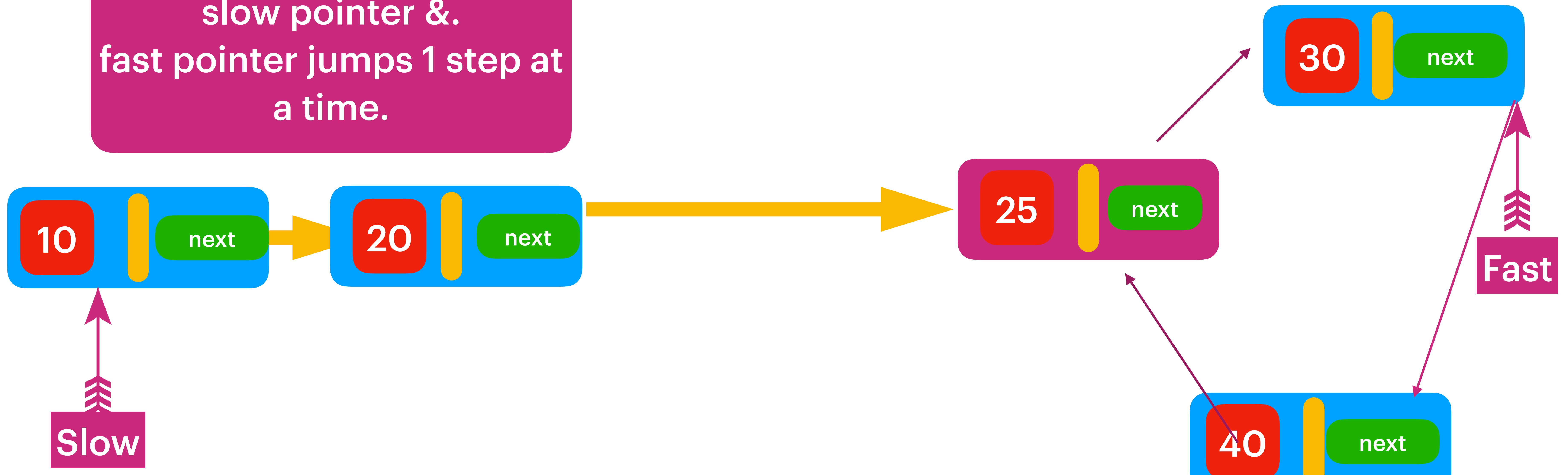


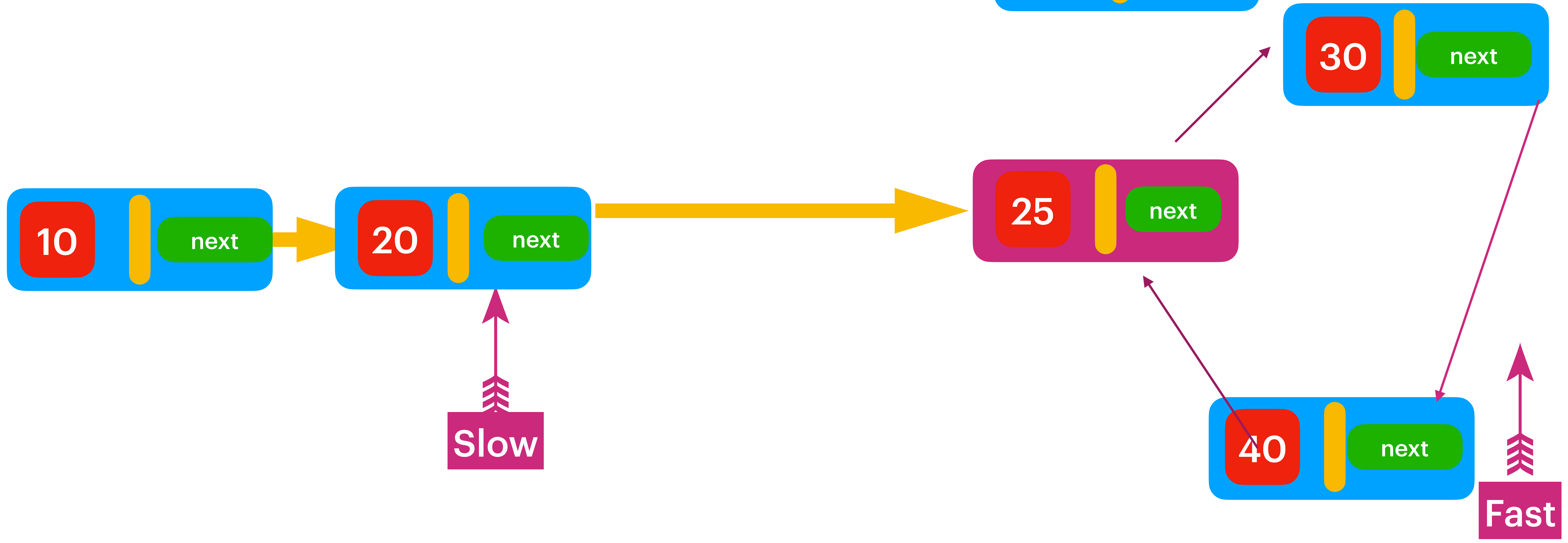
After reaching to loop, then reset Slow pointer head.
After then both the fast and slow pointers moves one step at a time.



After reset of slow Pointer to head !!!
slow pointer &.
fast pointer jumps 1 step at a time.

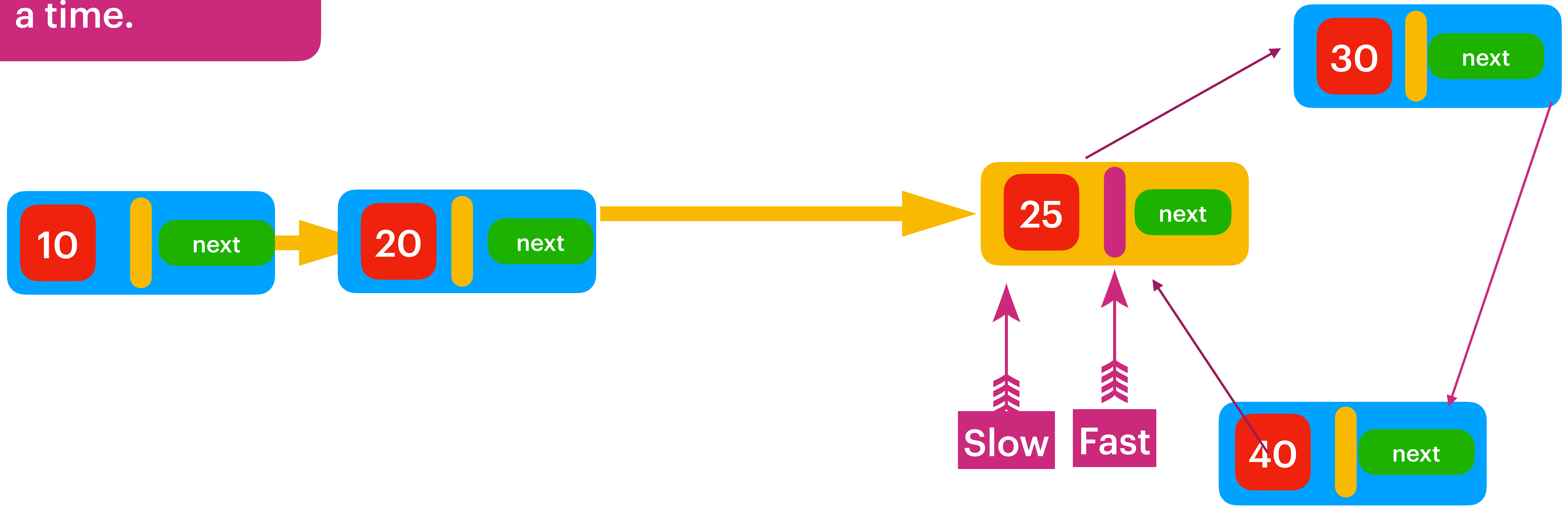
FindOut Loop Starting point





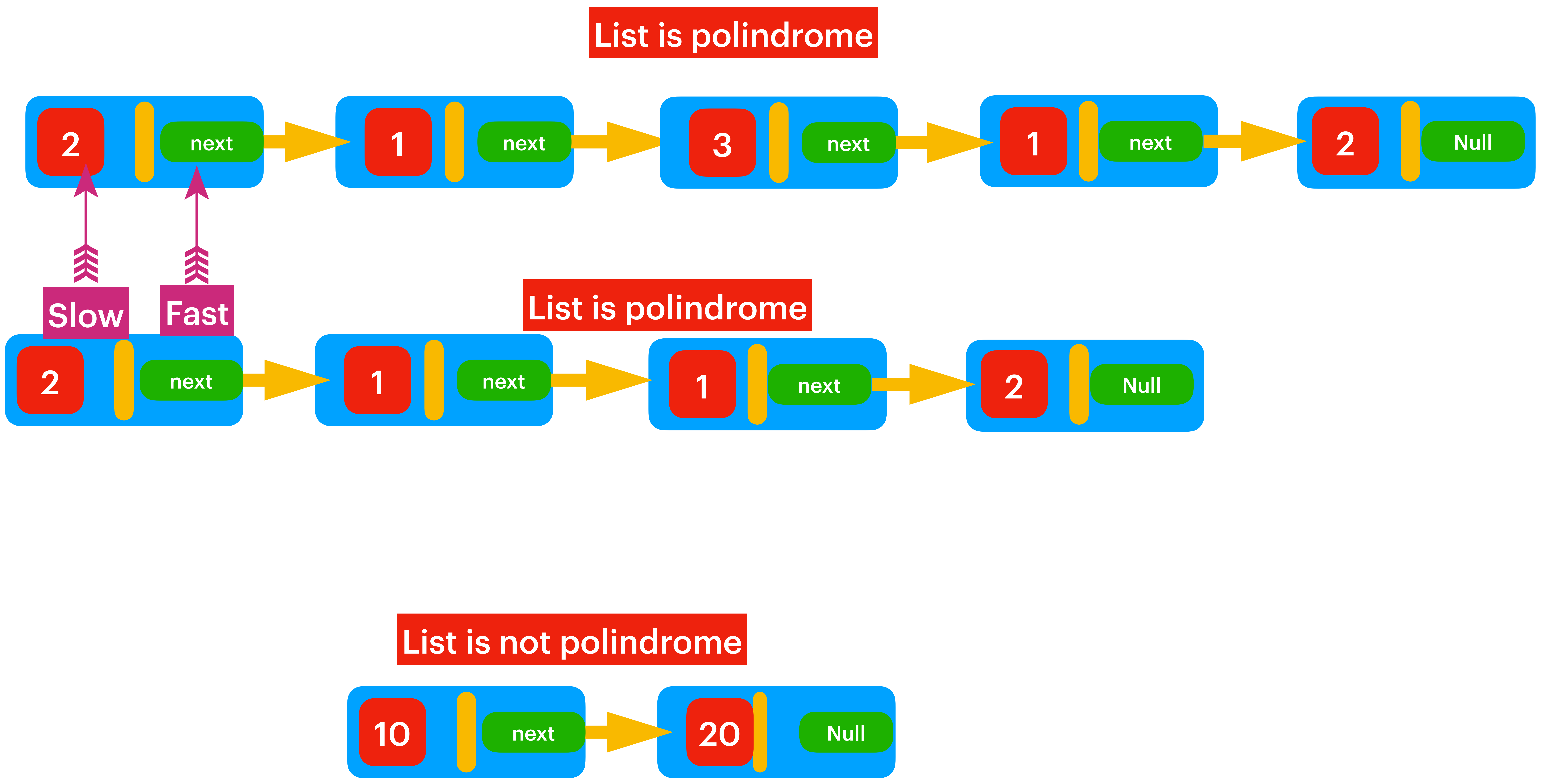
After reset of slow Pointer to head !!!
slow pointer &.
fast pointer jumps 1 step at a time.

FindOut Loop Starting point

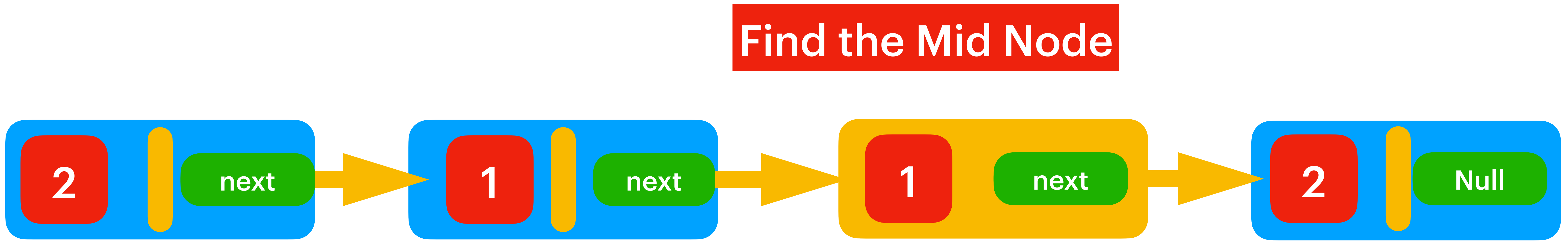


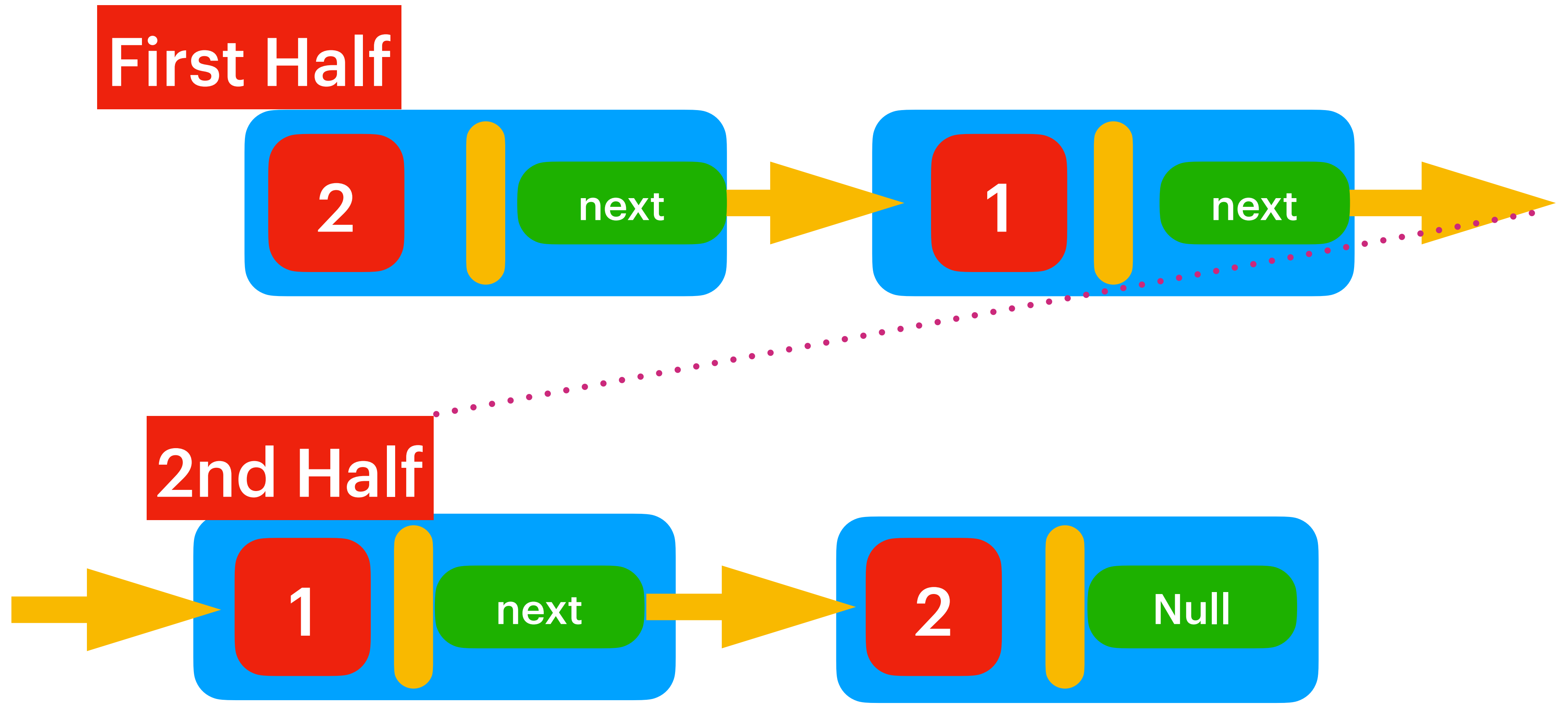
When Slow == Fast then that is going to be the loop starting point.

Palindrome Linked List

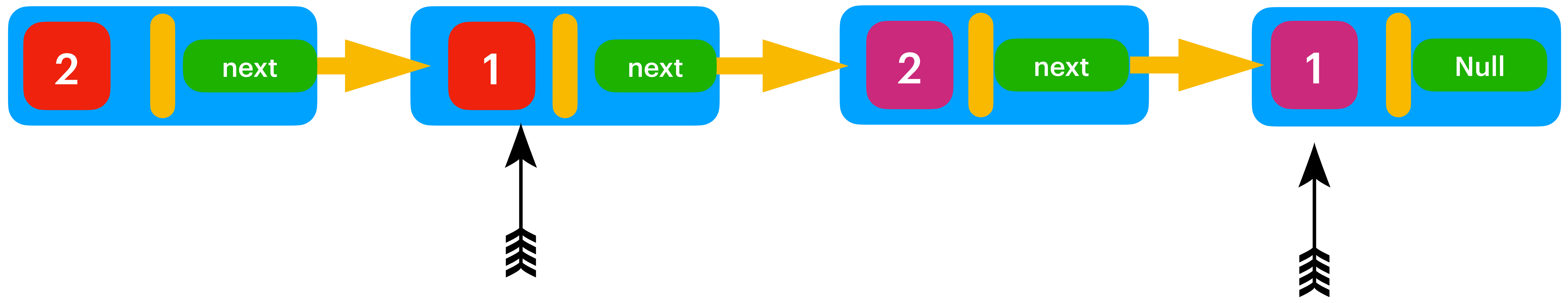


Palindrome Linked List



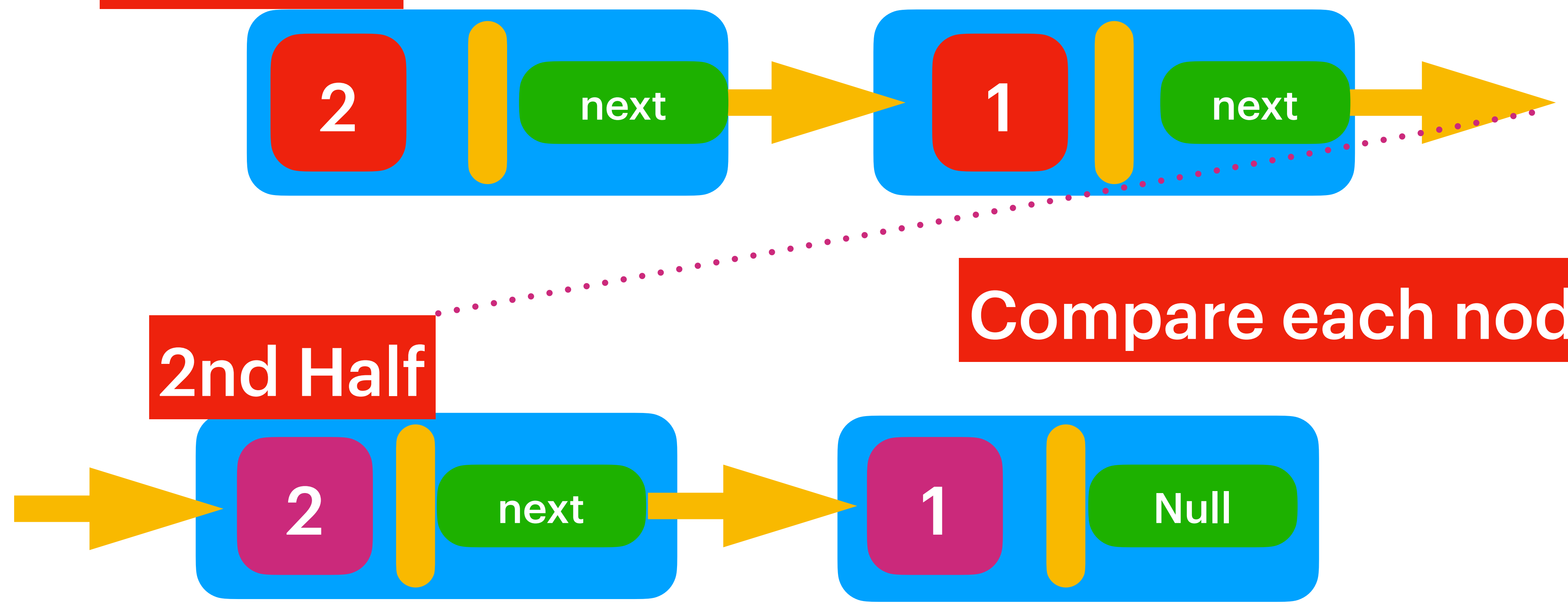


Reverse the 2ndHalf

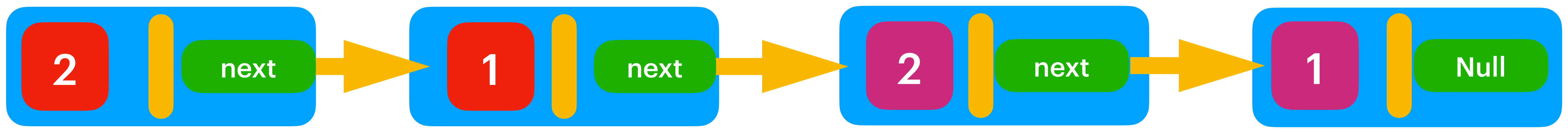


First Half

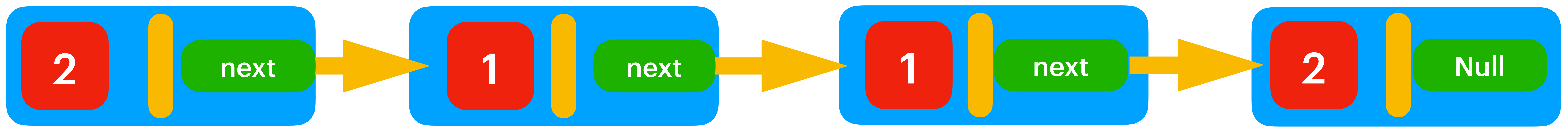
2nd Half



Compare each node from 1st Half & 2nd Half !!! Check does they are equal



Reverse the 2ndHalf to get back to original List


2
next
1
next
1
next
2
Null