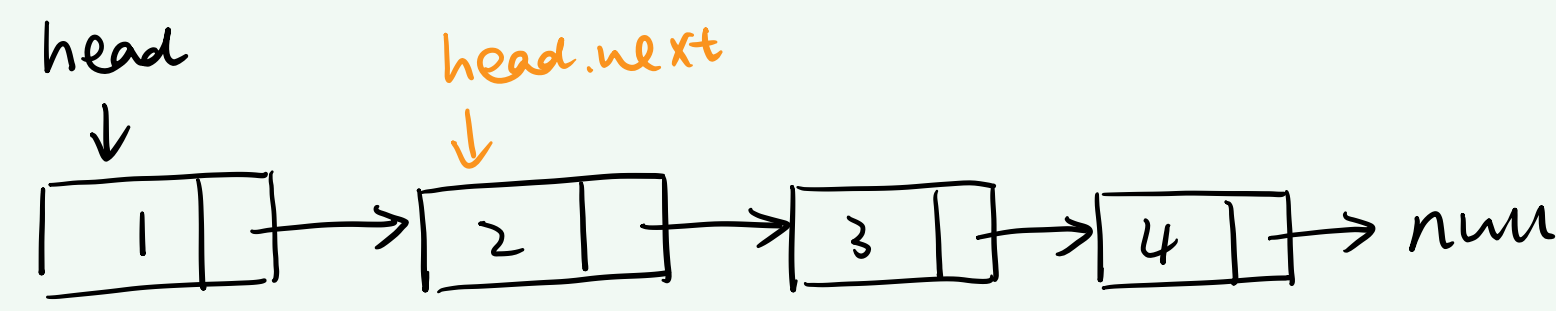


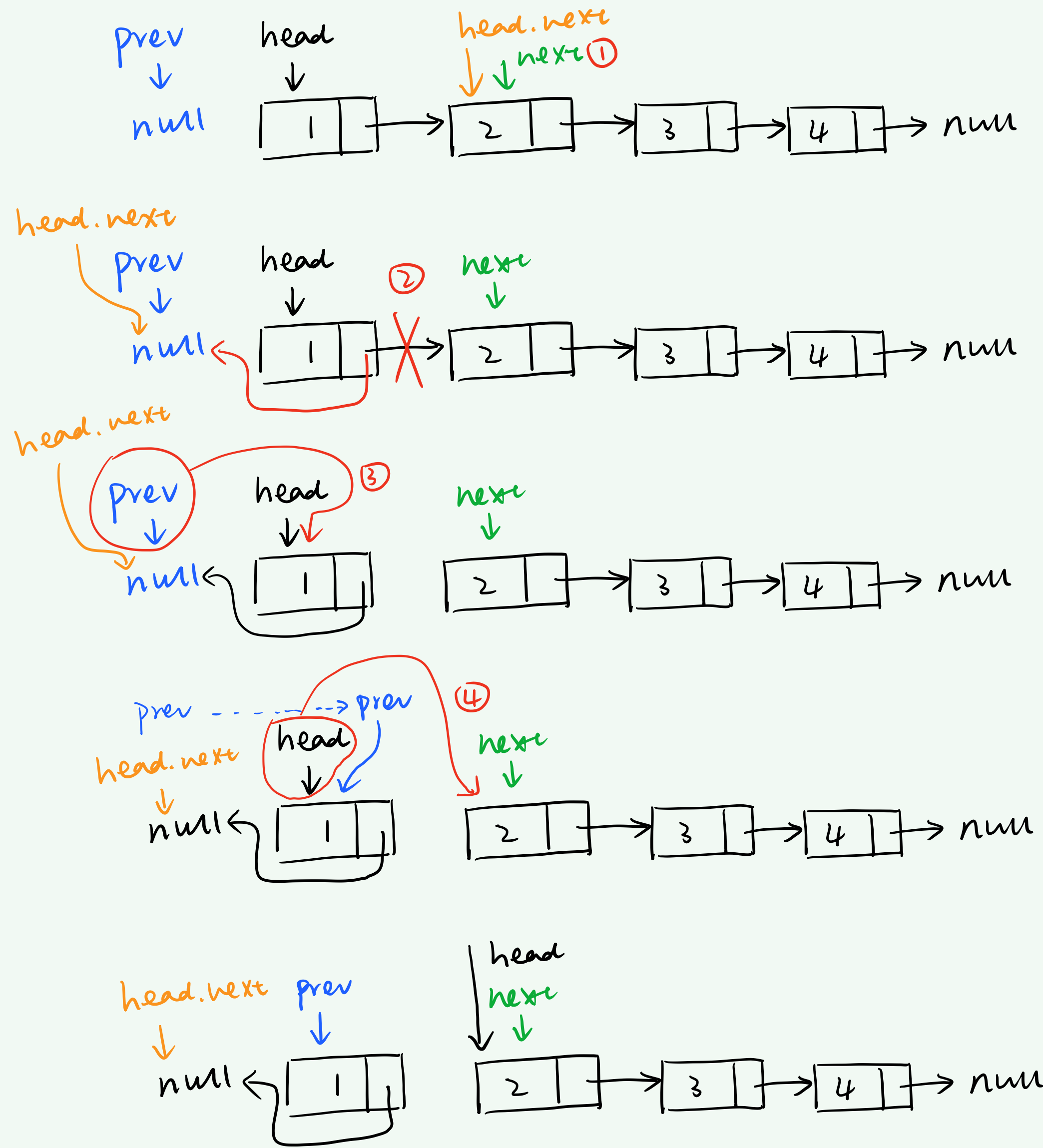
Reverse LinkedList\_Version 1\_Iterative



ListNode prev = null;

ListNode next = null;

while ( head.next != null ) {



}  
return prev;

① 给 head.next 找替身

next = head.next;

② 断开原来的 head → head.next 的指向, 令 head.next 指向 prev.

head.next = prev;

③ keep prev moving forward to do traversal of the LinkedList, prev's next node is "head"

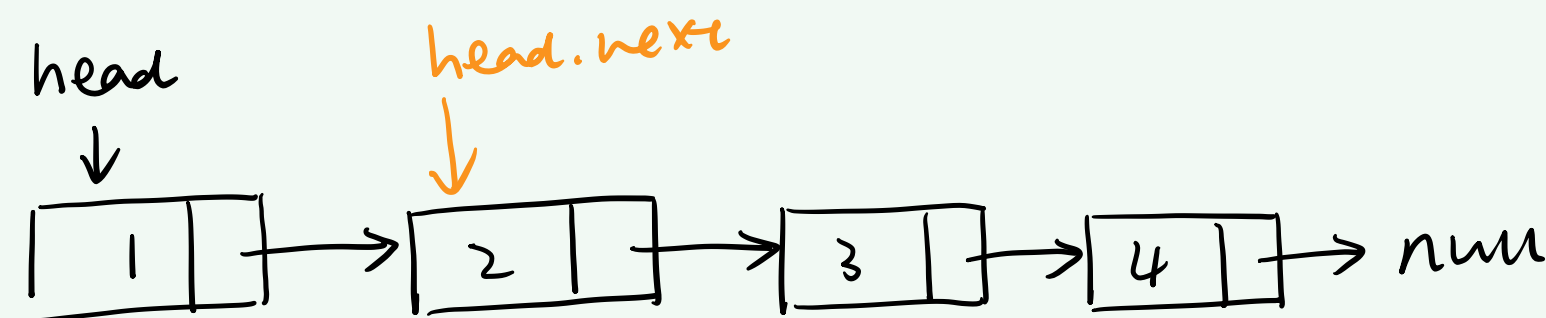
prev = head;

④ keep head moving forward to do traversal of the LinkedList, head's next node is "next"

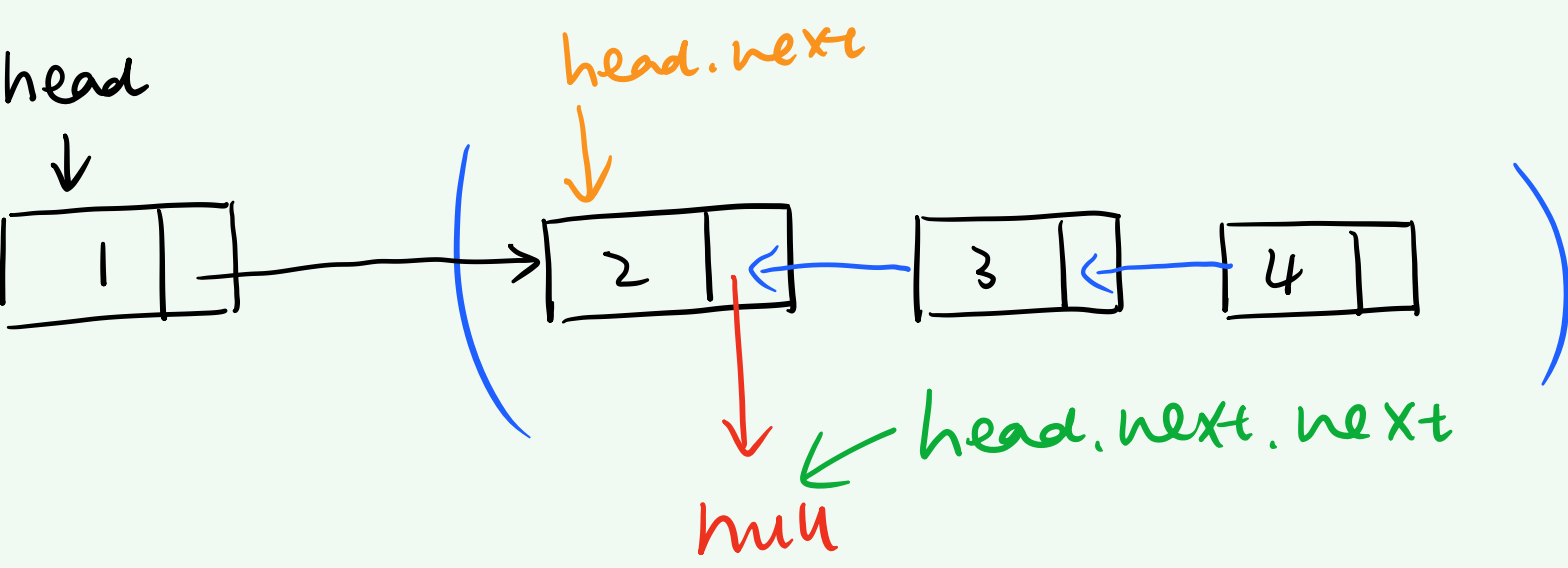
head = next;

这 4 步 "环环相扣"

Reverse LinkedList\_Version 2\_Recursive



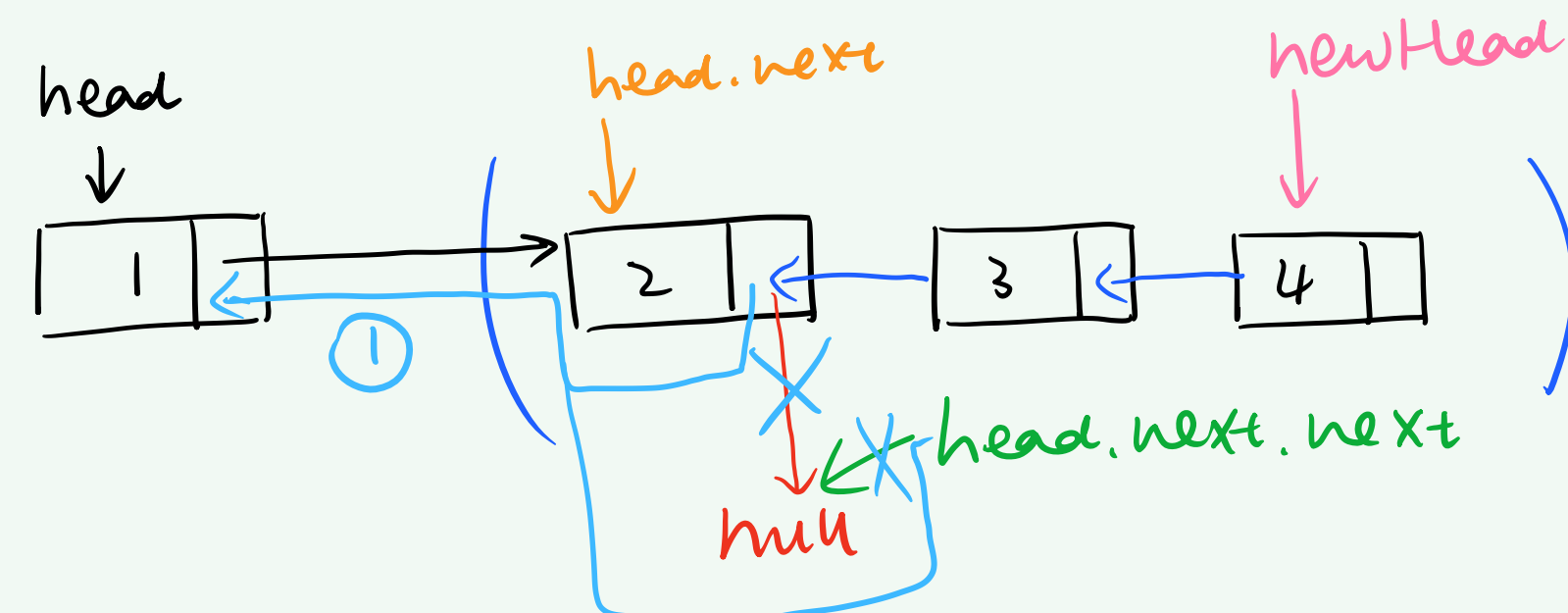
假设 head 后面的部分都已经 reverse 好了.



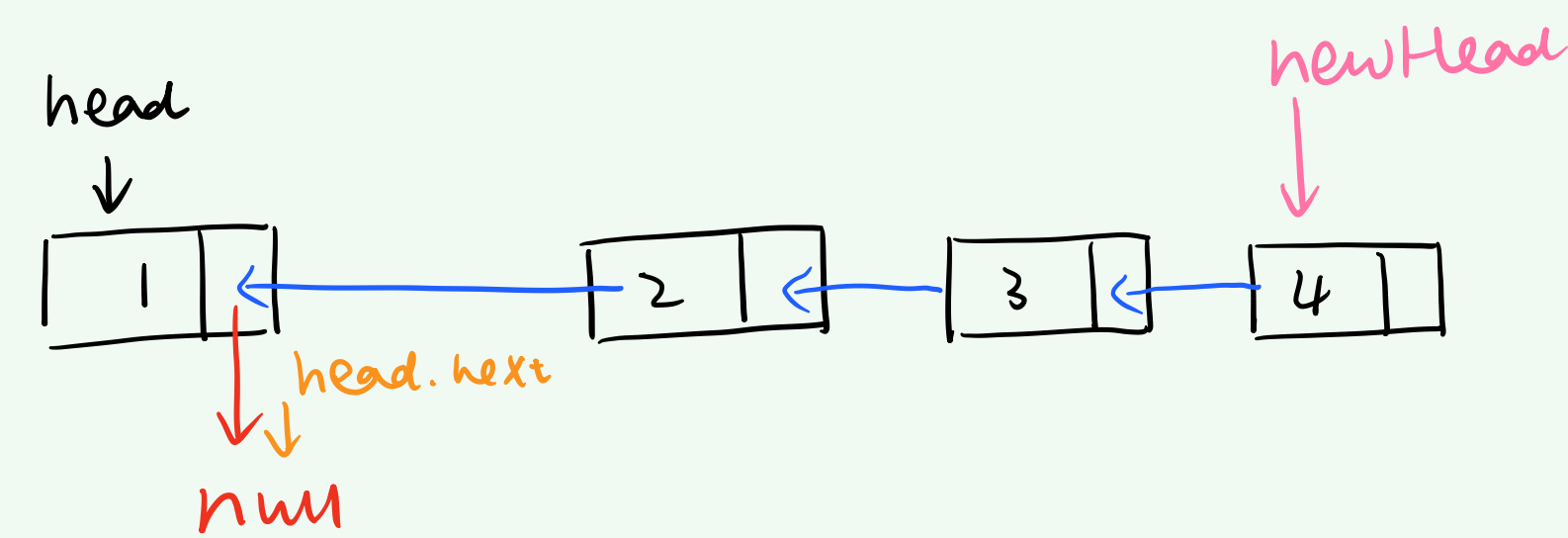
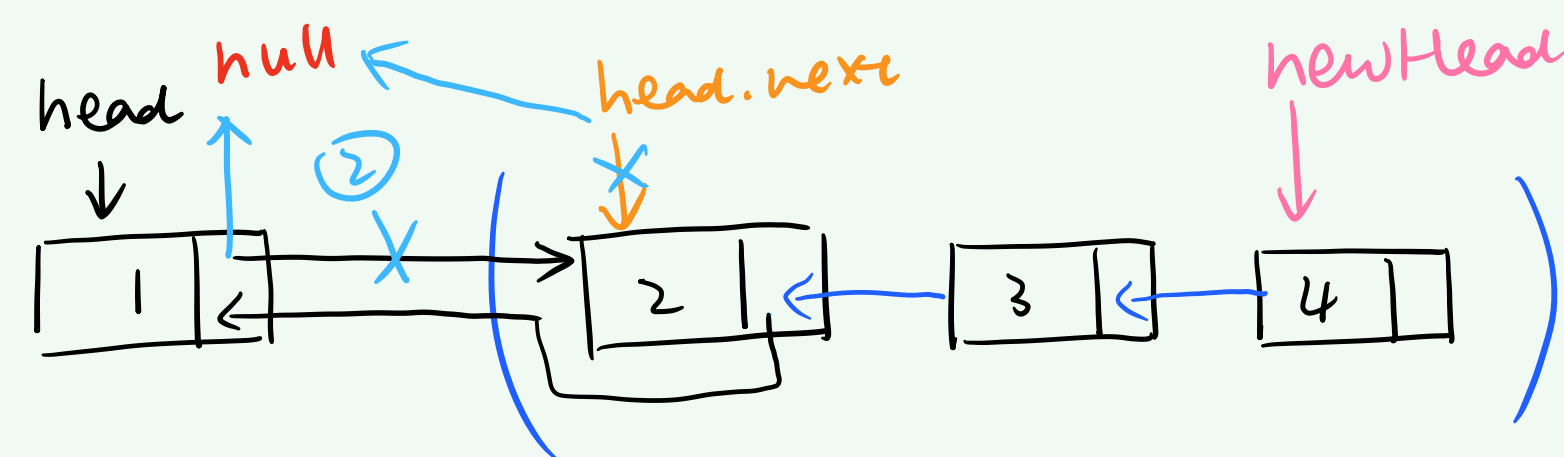
此时, head 指向 "1", 且 head.next 指向 2, 但 head.next.next 指向 null

要做的事:

① 让 head.next.next 指向 head



② 让 head.next 指向 null



③ return newHead