

极客学院
jikexueyuan.com

名企数据结构面试题之链表（中）

名企数据结构面试题之链表（中） — 课程概要

- 寻找链表的倒数第N个节点
- 删除链表的倒数第N个节点
- 查询节点并删除
- 删除排序链表的重复节点
- 删除排序链表的所有重复节点
- 反转链表的指定部分

寻找链表的倒数第N个节点

寻找链表的倒数第N个节点

- 问题描述
- 普通算法的思路
- 普通算法的实现
- OnePass算法的思路
- OnePass算法的实现

寻找链表的倒数第N个节点 — 问题描述

寻找一个单链表的倒数第N个节点，N从1开始计数。

样例输入：1→2→3→4→5→6，N = 4

样例输出：值为3的节点

假设N不会越界。

寻找链表的倒数第N个节点 — 普通算法的思路

思路1:

- 将指针移动到链表尾部null
- 将指针往前回退N位



单链表没有pre指针!

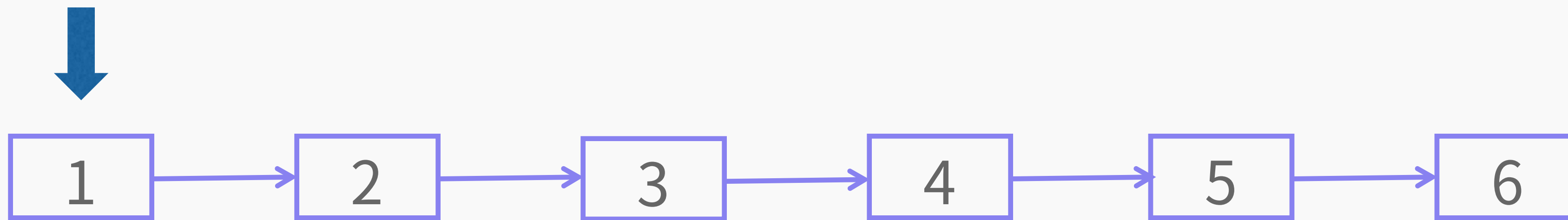
寻找链表的倒数第N个节点 — 普通算法的思路

思路2:

假设链表的长度为M，寻找倒数第N个节点，相当于寻找链表的第M-N+1个节点。

M = 6, N = 4, 那么M-N+1 = 3。

- 遍历链表，取得链表的长度M
- 再次遍历链表，找到第M-N+1个节点



注意：第二次遍历，指针向后移动M-N次。

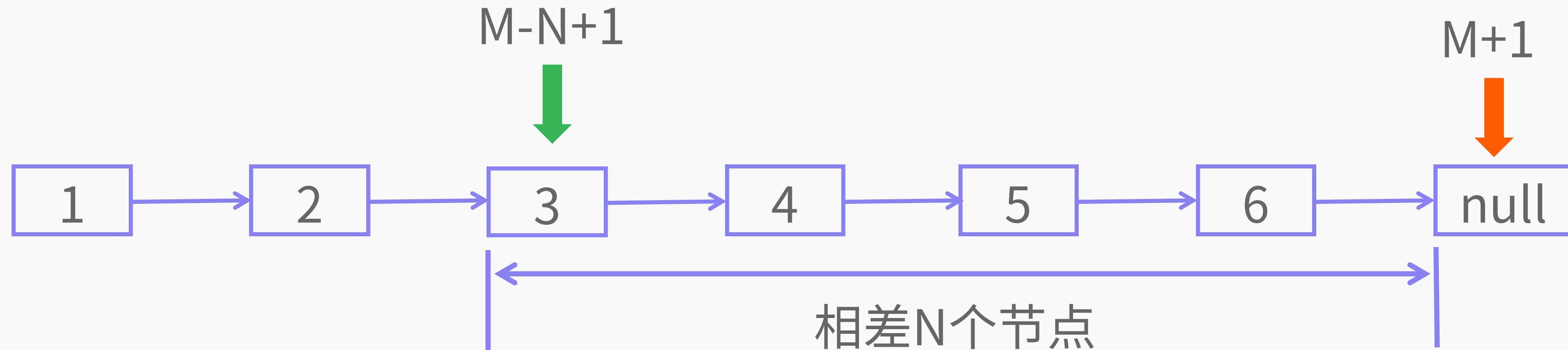
寻找链表的倒数第N个节点 — 普通算法的实现

Key	Value
类名	NthNodeFromEndOfList
方法名	find01
时间复杂度	O(M)，M表示链表长度
空间复杂度	O(1)
测试输入	1→2→3→4→5→6，N = 4
测试输出	3

寻找链表的倒数第N个节点 — OnePass算法的思路

额外要求：

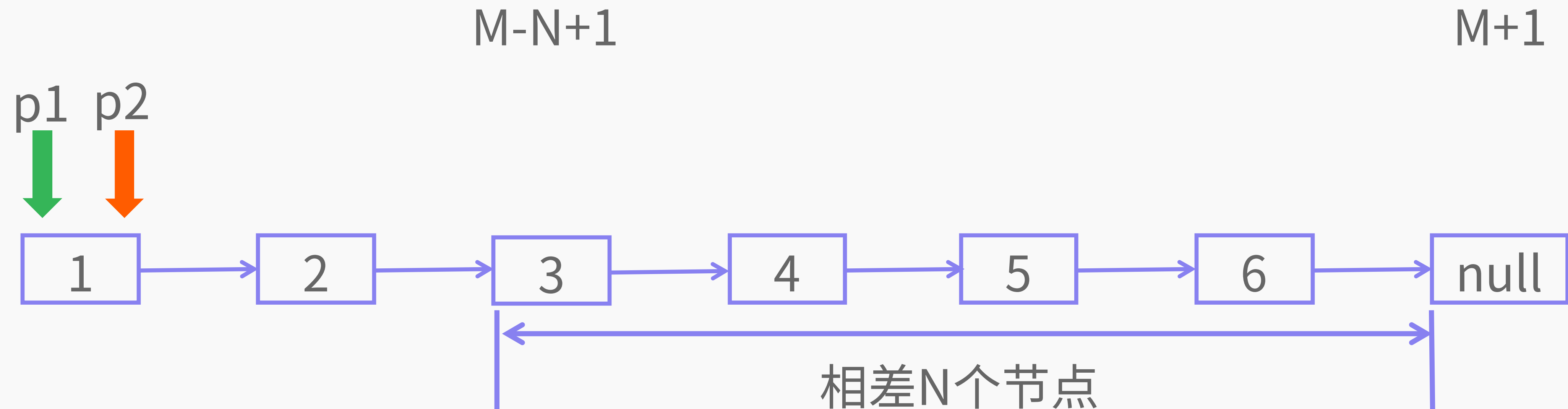
- 只允许遍历一次链表，也就是OnePass
- 允许存在多个指针



寻找链表的倒数第N个节点 — OnePass算法的思路

OnePass算法：

- 定义指针p1、p2
- 指针p2往后移动N位
- 同时将p1、p2往后移动，直到p2遇到null



寻找链表的倒数第N个节点 — OnePass算法的实现

Key	Value
类名	NthNodeFromEndOfList
方法名	find02
时间复杂度	O(M)，M表示链表长度
空间复杂度	O(1)
测试输入	1→2→3→4→5→6，N = 4
测试输出	3

删除链表的倒数第N个节点

删除链表的倒数第N个节点

- 问题描述
- 思路分析
- 代码实现
- 测试与提交

删除链表的倒数第N个节点 — 问题描述

leetcode 19: Remove Nth Node From End Of List

给定单链表，删除单链表的倒数第N个节点，N从1开始计数。

样例输入：1→2→3→4→5→6，N = 4

样例输出：1→2→4→5→6

额外要求：

- InPlace；空间复杂度必须为 $O(1)$
- OnePass；只允许遍历一次链表，但允许多个指针

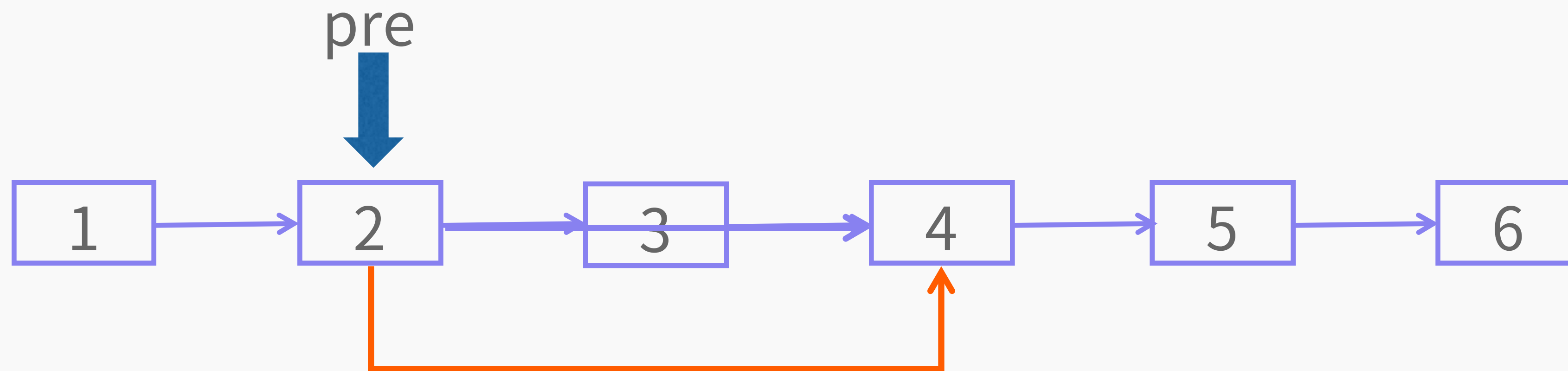
删除链表的倒数第N个节点 — 思路分析

删除节点p，关键步骤？

找到p的前趋节点pre。

删除链表的倒数第N个节点：

- 找到链表的倒数第N+1个节点pre
- `pre.next=pre.next.next`



删除链表的倒数第N个节点 — 代码实现

特殊情况：删除头结点

Key	Value
类名	_19RemoveNthNodeFromEndOfList
方法名	removeNthFromEnd
时间复杂度	O(M)，M表示链表长度
空间复杂度	O(1)

删除链表的倒数第N个节点 — 测试与提交

Key	Value
类名	_19RemoveNthNodeFromEndOfList
方法名	test
测试输入1	1→2→3→4→5→6, N = 4
测试输出1	1→2→4→5→6
测试输入2	1→2→3→4→5, N = 2
测试输出2	1→2→3→5

查询节点并删除

查询节点并删除

- 问题描述
- 思路分析
- 代码实现
- 测试与提交

查询节点并删除 — 问题描述

leetcode 203: Remove Linked List Elements

给定单链表、值val，删除所有值为val的节点。

样例输入：1→6→4→6→6→3→5，val = 6

样例输出：1→4→3→5

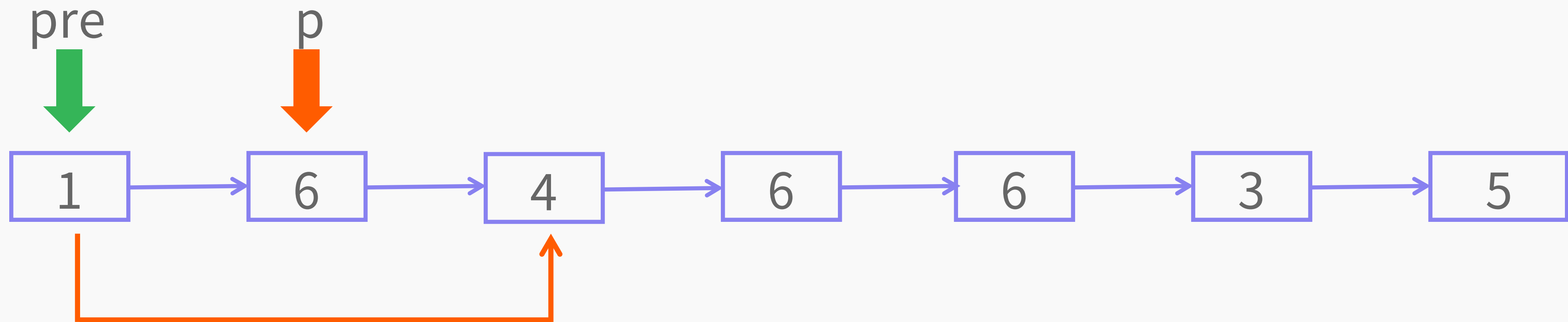
额外要求：

- InPlace
- OnePass

查询节点并删除 — 思路分析

删除，pre指针、p指针。

- 如果p的值等于val，删除；pre不动，p往后移动一位
- 否则，pre、p都往后移动一位



边界情况：头结点head的值为val。

查询节点并删除 — 代码实现

Key	Value
类名	_203RemoveLinkedListElements
方法名	removeElements
时间复杂度	$O(N)$ ，N表示链表长度
空间复杂度	$O(1)$

查询节点并删除 — 测试与提交

Key	Value
类名	_203RemoveLinkedListElements
方法名	test
测试输入1	1→6→6→4→6→3→5, val = 6
测试输出1	1→4→3→5
测试输入2	1→2→3→4→1, val = 1
测试输出2	2→3→4

删除排序链表的重复节点

删除排序链表的重复节点

- 问题描述
- 思路分析
- 代码实现
- 测试与提交

删除排序链表的重复节点 — 问题描述

leetcode 83: Remove Duplicates From Sorted List

给定已经排完序的单链表，删除重复元素，使得每个数字只出现一次。

样例输入：1→1→2→2→2→3→4

样例输出：1→2→3→4

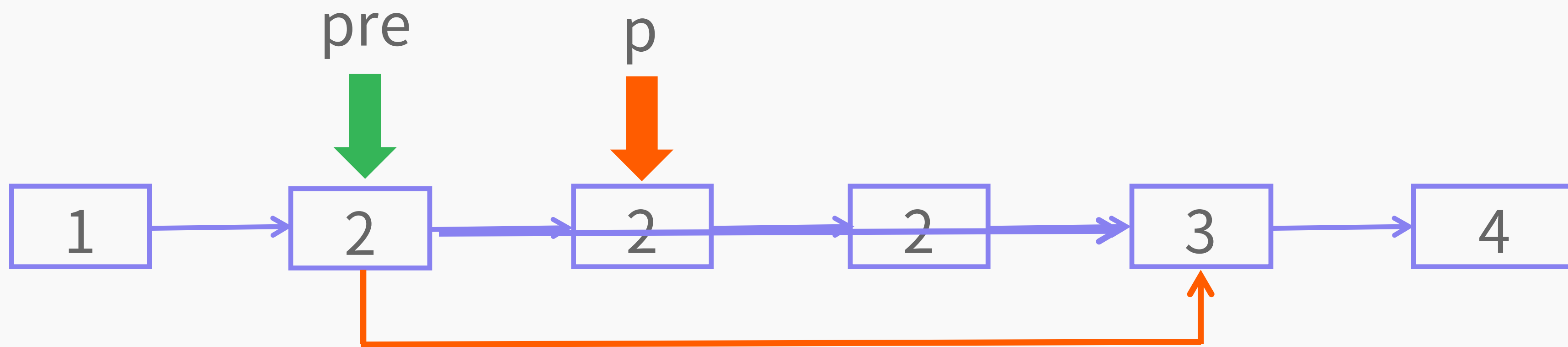
额外要求：

- InPlace
- OnePass

删除排序链表的重复节点 — 思路分析

删除，pre指针、p指针。

- 如果 $pre.val == p.val$ ，表明重复；p往后移动，直到不相等；删除操作
- 否则，不重复，pre、p往后移动一位



删除排序链表的重复节点 — 代码实现

Key	Value
类名	_083RemoveDuplicatesFromSortedList
方法名	deleteDuplicates
时间复杂度	$O(N)$ ，N表示链表长度
空间复杂度	$O(1)$

删除排序链表的重复节点 — 测试与提交

Key	Value
类名	_083RemoveDuplicatesFromSortedList
方法名	test
测试输入1	1→1→2→2→2→3→4
测试输出1	1→2→3→4
测试输入2	1→1→2→3→3
测试输出2	1→2→3

删除排序链表的所有重复节点

删除排序链表的所有重复节点

- 问题描述
- 思路分析
- 代码实现
- 测试与提交
- 解题技巧总结

删除排序链表的所有重复节点 — 问题描述

leetCode 82: Remove Duplicates from Sorted List II

给定已经排完序的单链表，删除含有重复元素的节点，只保留原本不重复的节点。

样例输入：1→2→3→3→3→4→4→5

样例输出：1→2→5

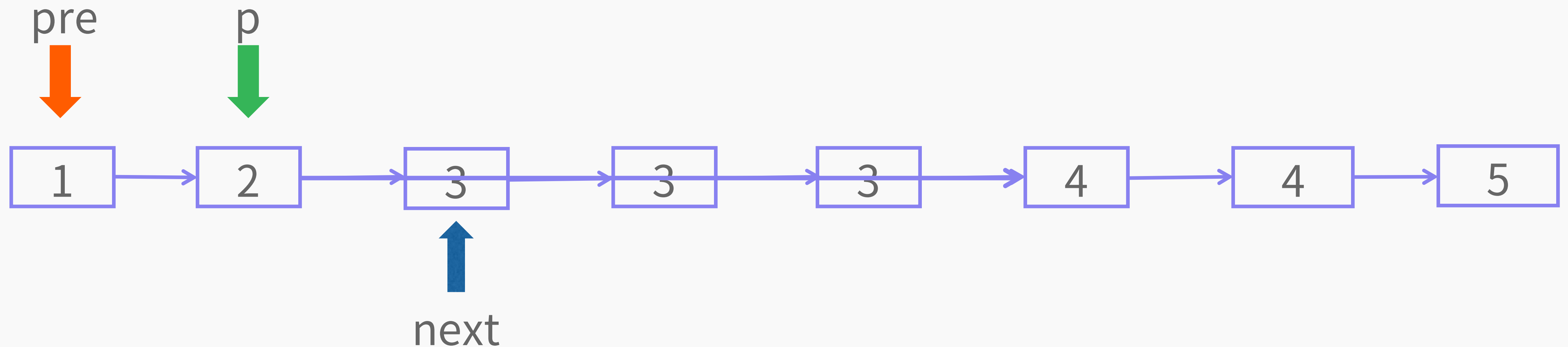
额外要求：

- InPlace
- OnePass

删除排序链表的所有重复节点 — 思路分析

删除自身，pre指针、p指针、next指针。

- $p.val \neq next.val$ ，不重复，pre、p往后移动一位
- $p.val == next.val$ ，表明重复；next往后移动，直到不相等；删除节点； $p = next$



删除排序链表的所有重复节点 — 代码实现

Key	Value
类名	_082RemoveDuplicatesFromSortedListII
方法名	deleteDuplicates
时间复杂度	O(N)，N表示链表长度
空间复杂度	O(1)

删除排序链表的所有重复节点 — 测试与提交

Key	Value
类名	_082RemoveDuplicatesFromSortedListII
方法名	test
测试输入1	1→2→3→3→3→4→4→5
测试输出1	1→2→5
测试输入2	1→1→1→2→3
测试输出2	2→3

删除排序链表的所有重复节点 — 解题技巧总结

删除链表节点的常用策略和技巧：

- 遍历链表、删除节点、调整指针指向
- 双指针、三指针
- 尽量OnePass、InPlace
- newHead，处理头结点

反转链表的指定部分

反转链表的指定部分

- 问题描述
- 思路分析
- 代码实现
- 测试与提交

反转链表的指定部分 — 问题描述

leetCode 92: Reverse Linked List II

反转单链表的某一部分。

$1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7$, $m = 3$, $n = 6$

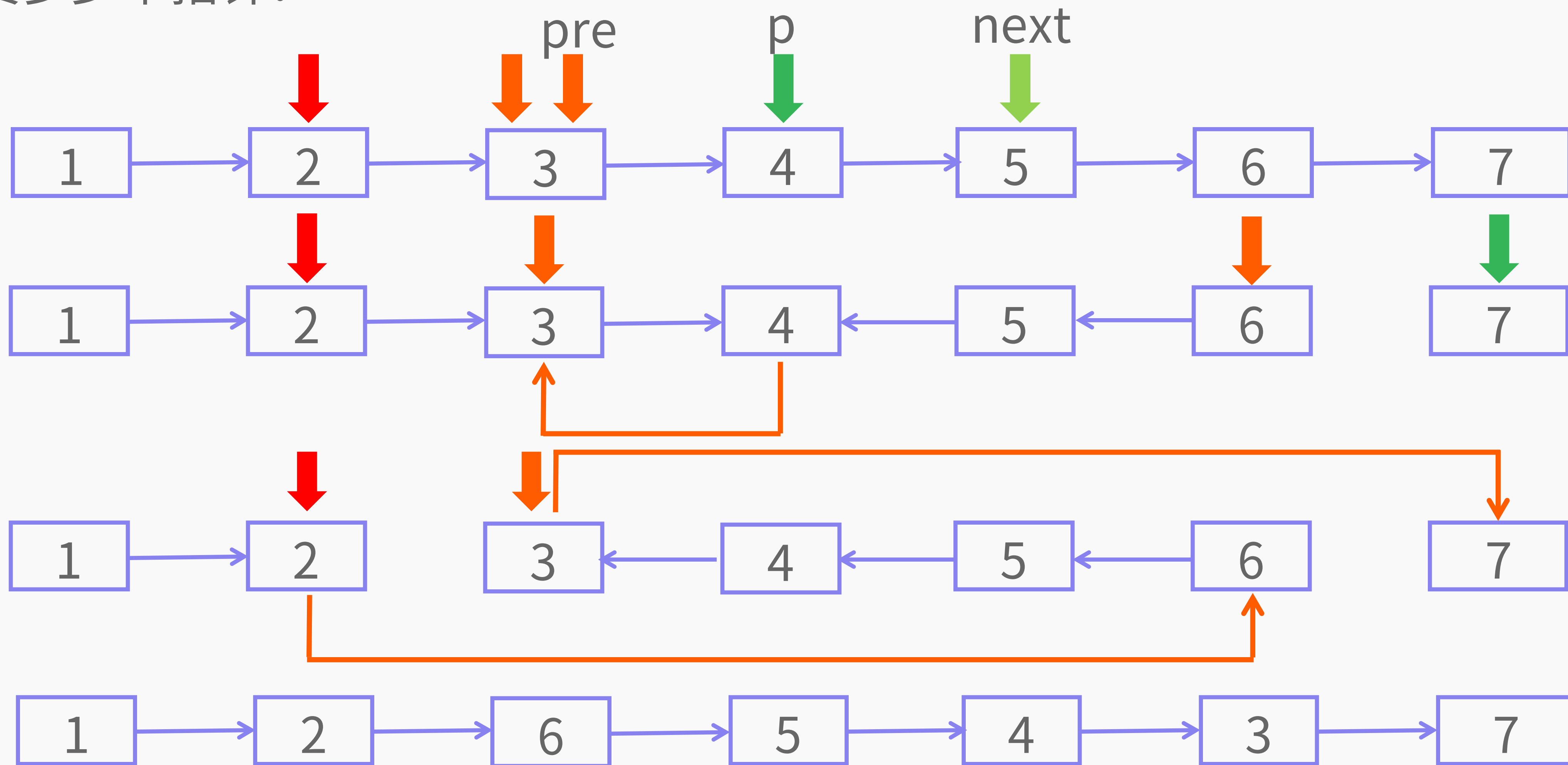
反转之后: $1 \rightarrow 2 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 7$

额外要求:

- InPlace
- OnePass

反转链表的指定部分 — 思路分析

需要多少个指针？



反转链表的指定部分 — 代码实现

Key	Value
类名	_092ReverseLinkedListII
方法名	reverseBetween
时间复杂度	$O(N)$ ，N表示链表长度
空间复杂度	$O(1)$

反转链表的指定部分 — 测试与提交

Key	Value
类名	_092ReverseLinkedListII
方法名	test
测试输入1	1→2→3→4→5→6→7, m = 3, n = 6
测试输出1	1→2→6→5→4→3→7
测试输入2	1→2→3→4→5→6→7, m = 1, n = 3
测试输出2	3→2→1→4→5→6→7

名企数据结构面试题之链表（中）

本套课程中我们学习了名企数据结构面试题之链表（中）。你应当掌握了以下知识：

- 寻找链表的倒数第N个节点
- 删除链表的倒数第N个节点
- 查询节点并删除
- 删除排序链表的重复节点
- 删除排序链表的所有重复节点
- 反转链表的指定部分

你可以使用leetcode验证程序是否正确，还可以在白纸上书写代码；如果想进一步提高，你可以继续在极客学院学习**名企数据结构面试题之链表（下）**课程。

极客学院

jikexueyuan.com

中国最大的IT职业在线教育平台

