

141 环形链表

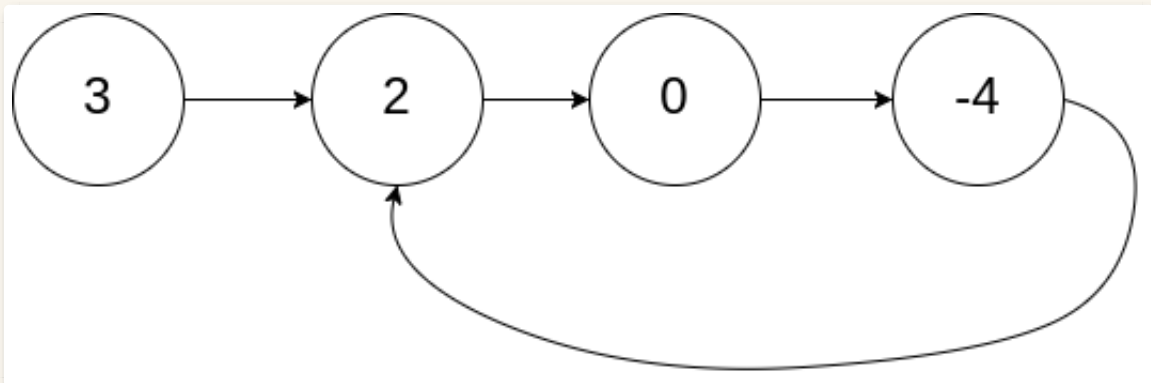
题目描述

给定一个链表，判断链表中是否有环。

为了表示给定链表中的环，我们使用整数 `pos` 来表示链表尾连接到链表中的位置（索引从 0 开始）。如果 `pos` 是 `-1`，则在该链表中没有环。

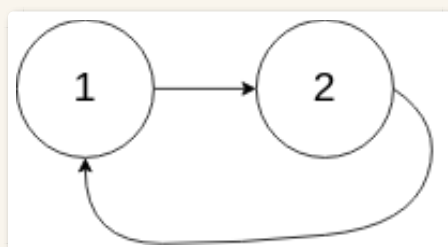
示例 1:

- 1 输入: `head = [3,2,0,-4]`, `pos = 1`
- 2 输出: `true`
- 3 解释: 链表中有一个环，其尾部连接到第二个节点。



示例 2:

- 1 输入: `head = [1,2]`, `pos = 0`
- 2 输出: `true`
- 3 解释: 链表中有一个环，其尾部连接到第一个节点。



示例 3:

```
1 输入: head = [1], pos = -1
2 输出: false
3 解释: 链表中没有环。
```



进阶:

你能用 $O(1)$ (即, 常量) 内存解决此问题吗?

代码

```
1  # Definition for singly-linked list.
2  # class ListNode(object):
3  #     def __init__(self, x):
4  #         self.val = x
5  #         self.next = None
6
7  class Solution(object):
8      def hasCycle(self, head):
9          """
10             :type head: ListNode
11             :rtype: bool
12             """
13             '''
14             快慢指针法, 如果存在环快指针一定会追上慢指针
15             '''
16             fast, slow = head, head
17             while fast and fast.next:
18                 fast = fast.next.next
19                 slow = slow.next
20                 if fast == slow:
21                     return True
22             return False
23
```

成功 [显示详情](#) >

执行用时：48 ms, 在Linked List Cycle的Python提交中击败了96.72% 的用户

内存消耗：18 MB, 在Linked List Cycle的Python提交中击败了33.83% 的用户

进行下一个挑战：

快乐数

炫耀一下：   

提交时间	状态	执行用时	内存消耗	语言
几秒前	通过	48 ms	18 MB	python