10月学习笔记leetcode

Day 1

[02.01.01 链表基础知识（第 01 ~ 02 天）](https://datawhalechina.github.io/leetcode-notes/#/ch02/02.01/02.01.01-Linked-List-Basic)

**「求线性链表长度」** 的代码如下：

# 获取链表长度

def length(self):

count = 0

cur = self.head

while cur:

count += 1

cur = cur.next

return count

**「链表尾部插入元素」** 的代码如下：

# 链表尾部插入元素

def insertRear(self, val):

node = ListNode(val)

cur = self.head

while cur.next:

cur = cur.next

cur.next = node

while的地方有不同之处，上面是要遍历到最后一个节点后停止。

下面要遍历到最后一个节点时一位停止

**「链表尾部删除元素」** 的代码如下：

# 链表尾部删除元素

def removeRear(self):

# 如果链表只有一个元素或没有元素，则返回错误

if not self.head.next:

return 'Error'

cur = self.head

while cur.next.next:

cur = cur.next

cur.next = None

此处while要遍历到最后一个节点前的一个节点停止

**「链表中间插入元素」** 的代码（自已用for写了一遍）如下：

def insertInside(self, index, val):

cur = self.head

count = 0

for count=0 in index-1

count += 1

cur = cur.next

# 这一步是检查指针变量是否还在链表，如果不在链表，返回空，not 返回true，执行return

if not cur:

return 'Error'

node = ListNode(val)

node.next = cur.next

cur.next = node

**「链表中间删除元素」** 的代码如下：

# 链表中间删除元素

def removeInside(self, index):

count = 0

cur = self.head

while cur.next and count < index - 1:

count += 1

cur = cur.next

if not cur:

return 'Error'

# 这里为什么不能直接cur.next = cur.next.next？

del\_node = cur.next

cur.next = del\_node.next

# 尝试用for改写

def removeInside(self, index):

count = 0

cur = self.head

for count in index-1

count += 1

cur = cur.next

if not cur

return ‘Error’

cur.next = cur.next.next