

剑指Offer（十四）：链表中倒数第k个结点

🕒 2017年12月1日 11:08:19 🗨 8 🌡 7,960 °C 🛠 编辑



一、前言

本系列文章为《剑指Offer》刷题笔记。

刷题平台：[牛客网](#)

书籍下载：[共享资源](#)

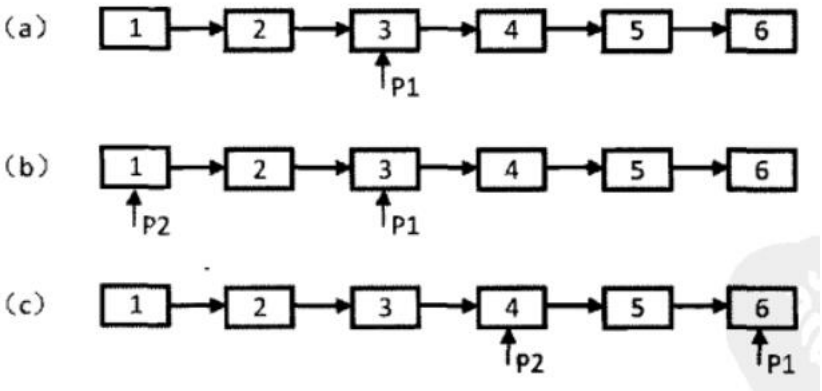
二、题目

输入一个链表，输出该链表中倒数第k个结点。

1、思路

我们可以定义两个指针。第一个指针从链表的头指针开始遍历向前走k-1，第二个指针保持不动；从第k步开始，第二个指针也开始从链表的头指针开始遍历。由于两个指针的距离保持在k-1，当第一个（走在前面的）指针到达链表的尾结点时，第二个指针（走在后面的）指针正好是倒数第k个结点。

效果示意图，以链表总共6个结点，求倒数第3个结点为例：



除此之外，要注意代码的鲁棒性。需要判断传入参数合法性问题。

2、代码

C++:

```
1  /*
2  struct ListNode {
3      int val;
4      struct ListNode *next;
5      ListNode(int x) :
6          val(x), next(NULL) {
7      }
8  };*/
9  class Solution {
10 public:
11     ListNode* FindKthToTail(ListNode* pListHead, unsigned int k) {
12         if(pListHead == NULL || k == 0){
13             return NULL;
14         }
15     }
```

```
15     ListNode *pAhead = pListHead;
16     ListNode *pBehind = pListHead;
17     for(unsigned int i = 0; i < k - 1; i++){
18         if(pAhead->next != NULL){
19             pAhead = pAhead->next;
20         }
21         else{
22             return NULL;
23         }
24     }
25     while(pAhead->next != NULL){
26         pAhead = pAhead->next;
27         pBehind = pBehind->next;
28     }
29     return pBehind;
30 }
31 };
```


Python2.7:

方法与C++一致：

	Pytho
<pre>1 # -*- coding:utf-8 -*- 2 # class ListNode: 3 # def __init__(self, x): 4 # self.val = x 5 # self.next = None 6 7 class Solution: 8 def FindKthToTail(self, head, k): 9 # write code here 10 if head == None or k == 0: 11 return None 12 phead = head 13 pbehind = head 14 for i in range(k-1): 15 if phead.next == None: 16 return None 17 else: 18 phead = phead.next 19 while phead.next != None: 20 phead = phead.next 21 pbehind = pbehind.next 22 return pbehind</pre>	

用列表，开辟新空间了，不太好。

	Pytho
<pre>1 # -*- coding:utf-8 -*- 2 # class ListNode: 3 # def __init__(self, x): 4 # self.val = x 5 # self.next = None 6 7 class Solution: 8 def FindKthToTail(self, head, k): 9 # write code here 10 l = [] 11 while head: 12 l.append(head) 13 head = head.next 14 if len(l) < k or k < 1: 15 return None 16 return l[-k]</pre>	



微信公众号

分享技术，乐享生活：微信公众号搜索「JackCui-AI」关注一个在互联网摸爬滚打的潜行者。

哀其不幸，怒其不争。--- 鲁迅