Rotate List

Given a list, rotate the list to the right by k places, where k is non-negative.

For example:
Given `1->2->3->4->5->NULL` and k = `2`,
return `4->5->1->2->3->NULL`.

Solution 1

There is no trick for this problem. Some people used slow/fast pointers to find the tail node but I don't see the benefit (in the sense that it doesn't reduce the pointer move op) to do so. So I just used one loop to find the length first.

```
class Solution {
public:
    ListNode* rotateRight(ListNode* head, int k) {
        if(!head) return head;

        int len=1; // number of nodes
        ListNode *newH, *tail;
        newH=tail=head;

        while(tail->next)  // get the number of nodes in the list
        {
            tail = tail->next;
            len++;
        }
        tail->next = head; // circle the link

        if(k %= len)
        {
            for(auto i=0; i<len-k; i++) tail = tail->next; // the tail node is th
e (len-k)-th node (1st node is head)
        }
        newH = tail->next;
        tail->next = NULL;
        return newH;
    }
};
```

written by dong.wang.1694 original link here

Solution 2

Since n may be a large number compared to the length of list. So we need to know the length of linked list.After that, move the list after the (l-n%l)th node to the front to finish the rotation.

Ex: {1,2,3} k=2 Move the list after the 1st node to the front

Ex: {1,2,3} k=5, In this case Move the list after (3-5%3=1)st node to the front.

So the code has three parts.

1) Get the length

2) Move to the (l-n%l)th node

3)Do the rotation

```
public ListNode rotateRight(ListNode head, int n) {
    if (head==null||head.next==null) return head;
    ListNode dummy=new ListNode(0);
    dummy.next=head;
    ListNode fast=dummy,slow=dummy;

    int i;
    for (i=0;fast.next!=null;i++)//Get the total length
        fast=fast.next;

    for (int j=i-n%i;j>0;j--) //Get the i-n%i th node
        slow=slow.next;

    fast.next=dummy.next; //Do the rotation
    dummy.next=slow.next;
    slow.next=null;

    return dummy.next;
}
```

written by reeclapple original link here

Solution 3

i am not getting that what i should do when K is greater than size of the list.

written by rforritz original link here

From Leetcoder.