

LEET CODE: Demonstration of program on Singly linked list

SOURCE CODE:

92. Reverse Linked List II

Medium Topics Companies

Given the head of a singly linked list and two integers *left* and *right* where *left* ≤ *right*, reverse the nodes of the list from position *left* to position *right*, and return the reversed list.

Example 1:

Input: head = [1,2,3,4,5], left = 2, right = 4
Output: [1,4,3,2,5]

Example 2:

Input: head = [5], left = 1, right = 1
Output: [5]

Constraints:

- The number of nodes in the list is *n*.
- 1 ≤ *n* ≤ 500
- 1 ≤ Node.val ≤ 500
- 1 ≤ left ≤ right ≤ *n*

```
1 struct ListNode* reverseBetween(struct ListNode* start, int a, int b)
2 {
3     if(a==b)
4         return start;
5     b=b-1;
6     struct ListNode *node1=NULL,*node2=NULL,*nodeb=NULL,*nodea=NULL,*ptr=start;
7     int c=a;
8     while(ptr!=NULL)
9     {
10         if(c==a-1)
11             nodeb=ptr;
12         else if(c==a)
13             nodea=ptr;
14         else if(c==b)
15             nodeb=ptr;
16         else if(c==b+1)
17             nodea=ptr;
18         ptr=ptr->next;
19         c++;
20     }
21     ptr=ptr->next;
22     struct ListNode*pre=nodea,"temp
23     ptr=start;
24     c=a;
25     while(ptr!=NULL)
26     {
27         if(c==a && c==b)
28             temp=ptr->next;
29             ptr->next=pre;
30             pre=ptr;
31             ptr=temp;
32         else if(c==b)
33         {
34             ptr->next=pre;
35             if(a==0)
36                 start=ptr;
37             return start;
38         }
39     }
40 }
```

92. Reverse Linked List II

Medium Topics Companies

Given the head of a singly linked list and two integers *left* and *right* where *left* ≤ *right*, reverse the nodes of the list from position *left* to position *right*, and return the reversed list.

Example 1:

Input: head = [1,2,3,4,5], left = 2, right = 4
Output: [1,4,3,2,5]

Example 2:

Input: head = [5], left = 1, right = 1
Output: [5]

Constraints:

- The number of nodes in the list is *n*.
- 1 ≤ *n* ≤ 500
- 1 ≤ Node.val ≤ 500
- 1 ≤ left ≤ right ≤ *n*

```
16 else if(c==b+1)
17 {
18     nodea=ptr;
19     break;
20 }
21 c++;
22 ptr=ptr->next;
23 }
24 struct ListNode*pre=nodea,"temp
25 ptr=start;
26 c=a;
27 while(ptr!=NULL)
28 {
29     if(c==a && c==b)
30     {
31         temp=ptr->next;
32         ptr->next=pre;
33         pre=ptr;
34         ptr=temp;
35     }
36     else if(c==b)
37     {
38         ptr->next=pre;
39         if(a==0)
40             start=ptr;
41         else
42             nodeb->next=ptr;
43         break;
44     }
45     else
46         ptr=ptr->next;
47     c++;
48     return start;
49 }
50 }
51 }
```

OUTPUT:

92. Reverse Linked List II

Medium Topics Companies

Given the head of a singly linked list and two integers *left* and *right* where *left* ≤ *right*, reverse the nodes of the list from position *left* to position *right*, and return the reversed list.

Example 1:

Input: head = [1,2,3,4,5], left = 2, right = 4
Output: [1,4,3,2,5]

Example 2:

Input: head = [5], left = 1, right = 1
Output: [5]

Accepted Runtime: 2 ms

Case 1 Case 2

Input

head = [1,2,3,4,5]

left = 2

right = 4

Output

[1,4,3,2,5]

Expected

[1,4,3,2,5]

Contribute a testcase