

92. Reverse Linked List II

[My Submissions](#)

QuestionEditorial Solution

Total Accepted: **76037** Total Submissions: **267159** Difficulty: **Medium**

Reverse a linked list from position m to n . Do it in-place and in one-pass.

For example:

Given 1->2->3->4->5->NULL, $m = 2$ and $n = 4$,

return 1->4->3->2->5->NULL.

Note:

Given m, n satisfy the following condition:

$1 \leq m \leq n \leq \text{length of list}$.

[Subscribe](#) to see which companies asked this question

```
/**
 * Definition for singly-linked list.
 * struct ListNode {
 *     int val;
 *     ListNode *next;
 *     ListNode(int x) : val(x), next(NULL) {}
 * };
 */
class Solution {
public:
    ListNode* reverseBetween(ListNode* head, int m, int n) {
        if(head == NULL || head->next == NULL) return head;
        ListNode dummy(-1);
        dummy.next = head;
        ListNode *prev = &dummy;
        for(int i=0;i<m-1;i++)
        {
            prev = prev->next;
        }
    }
}
```

```
ListNode *head2 = prev;
prev = head2->next;
ListNode *cur = prev->next;
for(int i=m;i<n;i++)
{
    prev->next = cur->next;
    cur->next = head2->next;
    head2->next = cur;
    cur = prev->next;
}
return dummy.next;
}
};
```