86. Partition List

My Submissions

QuestionEditorial Solution

Total Accepted: **70108** Total Submissions: **233181** Difficulty: **Medium**

Given a linked list and a value *x*, partition it such that all nodes less than *x* come before nodes greater than or equal to *x*.

You should preserve the original relative order of the nodes in each of the two partitions.

For example,

Given 1->4->3->2->5->2 and *x* = 3,

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

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* partition(ListNode* head, int x) {
        ListNode left_dummy(-1);
        ListNode right_dummy(-1);
        ListNode *cur = head;
        ListNode *left_cur = &left_dummy;
        ListNode *right_cur = &right_dummy;
        while(cur)
        {
            if(cur->val <x)
            {
                left_cur->next = cur;
                left_cur = cur;
            }else
            {
                right_cur->next = cur;
                right_cur = cur;
            }
            cur = cur->next;
        }
        left_cur->next = right_dummy.next;
        right_cur->next = NULL;
        return left_dummy.next;
    }
};
```