10) [**148. Sort List**](https://leetcode.com/problems/sort-list/)  
CODE  
class Solution {

public:

ListNode\* merge(ListNode\* l1, ListNode\* l2) {

ListNode\* dummy = new ListNode(0);

ListNode\* current = dummy;

while (l1 && l2) {

if (l1->val < l2->val) {

current->next = l1;

l1 = l1->next;

} else {

current->next = l2;

l2 = l2->next;

}

current = current->next;

}

current->next = l1 ? l1 : l2;

return dummy->next;

}

ListNode\* sortList(ListNode\* head) {

if (!head || !head->next) return head;

ListNode\* slow = head, \*fast = head, \*prev = nullptr;

while (fast && fast->next) {

prev = slow;

slow = slow->next;

fast = fast->next->next;

}

prev->next = nullptr;

ListNode\* left = sortList(head);

ListNode\* right = sortList(slow);

return merge(left, right);

}

};  
OUTPUT  
