143. Reorder List ★

Question | Editorial Solution | My Submissions (/problems/reorder-list/submissions/)

Total Accepted: **75375** Total Submissions: **313645** Difficulty: **Medium**

Given a singly linked list L: $L_0 \rightarrow L_1 \rightarrow \ldots \rightarrow L_{n-1} \rightarrow L_n$,

reorder it to: $L_0 \rightarrow L_n \rightarrow L_1 \rightarrow L_{n-1} \rightarrow L_2 \rightarrow L_{n-2} \rightarrow \ldots$

You must do this in-place without altering the nodes' values.

For example,

Given `{1,2,3,4}`, reorder it to `{1,4,2,3}`.

Subscribe (/subscribe/) to see which companies asked this question

Show Tags

Have you met this question in a real interview? Yes No

Discuss (https://leetcode.com/discuss/questions/oj/reorder-list)

Pick One (/problems/random-one-question/)

Notes

C++

```
/**
 * Definition for singly-linked list.
 * struct ListNode {
 *     int val;
 *     ListNode *next;
 *     ListNode(int x) : val(x), next(NULL) {}
 * };
 */
class Solution {
public:
    void reorderList(ListNode* head) {
        if (head) {
            vector<ListNode*> cache;
            for (auto it = head; it != NULL; it = it->next) {
                cache.push_back(it);
            }
            ListNode dummy(0);
            ListNode* nowPtr = &dummy;
            auto first = cache.begin(), last = cache.end() - 1;
            for (; last > first && first != last; ++first) {
                nowPtr->next = *first;
                nowPtr = nowPtr->next;
                nowPtr->next = *last;
                nowPtr = nowPtr->next;
                --last;
```

```
26              }
27              if (first == last) {
28                  nowPtr->next = *first;
29                  nowPtr = nowPtr->next;
30              }
31              nowPtr->next = NULL;
32          }
33      }
```

Custom Testcase

Run Code | Submit Solution | Notes