23. Merge k Sorted Lists

Description Hints Submissions Solutions

- Total Accepted: **147945**
- Total Submissions: **550960**
- Difficulty: **Hard**
- Contributor: **LeetCode**

Merge *k* sorted linked lists and return it as one sorted list. Analyze and describe its complexity.

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:
    struct compare {
    bool operator()(const ListNode* l, const ListNode* r) {
        return l->val > r->val;
    }
};
ListNode *mergeKLists(vector<ListNode *> &lists) { //priority_queue
    priority_queue<ListNode *, vector<ListNode *>, compare> q;
    for(auto l : lists) {
        if(l)  q.push(l);
    }
    if(q.empty())  return NULL;

    ListNode* result = q.top();
    q.pop();
    if(result->next) q.push(result->next);
    ListNode* tail = result;
    while(!q.empty()) {
        tail->next = q.top();
```

```
        q.pop();
        tail = tail->next;
        if(tail->next) q.push(tail->next);
    }
    return result;
}
};
```