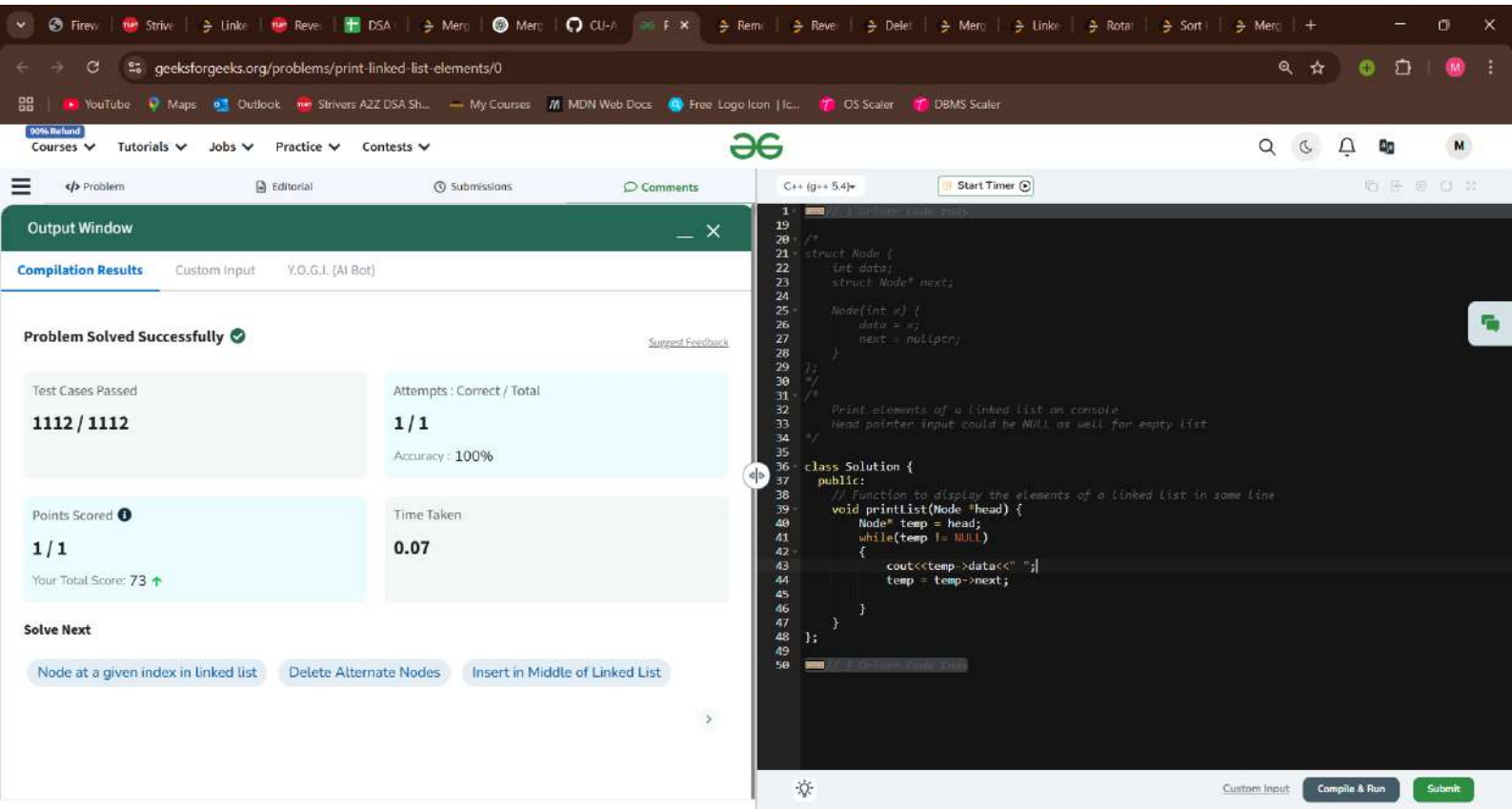

geeksforgeeks.org/problems/print-linked-list-elements/0
Courses
Tutorials
Jobs
Practice
Contests
Problem
Editorial
Submissions
Comments
Output Window
Compilation Results
Custom Input
Y.O.G.I. (AI Bot)
Problem Solved Successfully
Suggest Feedback
Test Cases Passed
1112 / 1112
Attempts : Correct / Total
1 / 1
Accuracy : 100%
Points Scored
1 / 1
Your Total Score: 73
Time Taken
0.07
Solve Next
Node at a given index in linked list
Delete Alternate Nodes
Insert in Middle of Linked List
C++ (g++ 5.4)
Start Timer
// } Driver Code Ends
/*
struct Node {
int data;
struct Node* next;
Node(int x) {
data = x;
next = nullptr;
}
};
*/
/*
Print elements of a linked list on console
Head pointer input could be NULL as well for empty list
*/
class Solution {
public:
// Function to display the elements of a linked list in same line
void printList(Node *head) {
Node* temp = head;
while(temp != NULL)
{
cout<<temp->data<<" ";
temp = temp->next;
}
}
};
// } Driver Code Ends
Custom Input
Compile & Run
Submit

leetcode.com/problems/remove-duplicates-from-sorted-list/submissions/1564877366/

YouTube | Maps | Outlook | Strivers A2Z DSA Sh... | My Courses | MDN Web Docs | Free Logo Icon | Ic... | OS Scaler | DBMS Scaler

Problem List | Run | Submit | Premium

Description | Editorial | Solutions | Submissions

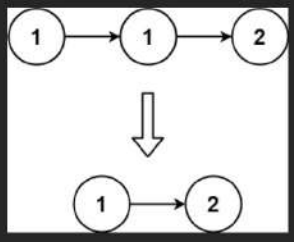
83. Remove Duplicates from Sorted List

Solved

Easy | Topics | Companies

Given the `head` of a sorted linked list, *delete all duplicates such that each element appears only once*. Return *the linked list* ***sorted*** *as well*.

Example 1:




```
Input: head = [1,1,2]
Output: [1,2]
```

Example 2:

9.1K | 108 | 80 Online

Code | Accepted

All Submissions

Accepted 168 / 168 testcases passed

Mayank Khandelwal submitted at Mar 06, 2025 18:30

Editorial | Solution

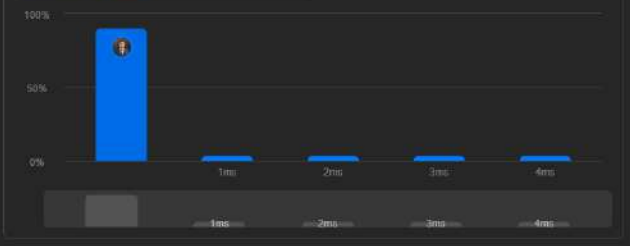
Runtime

0 ms | Beats 100.00%

Analyze Complexity

Memory

16.20 MB | Beats 67.73%



Testcase | Test Result

Accepted Runtime: 0 ms

Case 1 | Case 2

Input

leetcode.com/problems/reverse-linked-list/description/

YouTube | Maps | Outlook | Strivers A2Z DSA Sh... | My Courses | MDN Web Docs | Free Logo Icon | Ic... | OS Scaler | DBMS Scaler

Problem List | Run | Submit | Premium

Description | Editorial | Solutions | Submissions

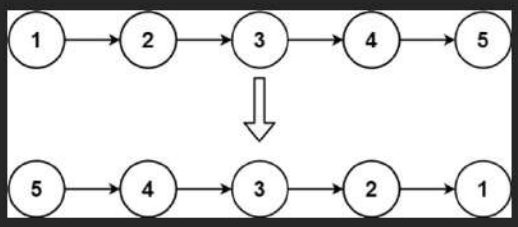
206. Reverse Linked List

Solved

Easy | Topics | Companies

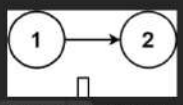
Given the `head` of a singly linked list, reverse the list, and return *the reversed list*.

Example 1:



Input: head = [1,2,3,4,5]
Output: [5,4,3,2,1]

Example 2:



22.6K | 272 | 332 Online

Code | Accepted

All Submissions

Accepted 28 / 28 testcases passed

Mayank Khandelwal submitted at Mar 06, 2025 15:37

Editorial | Solution

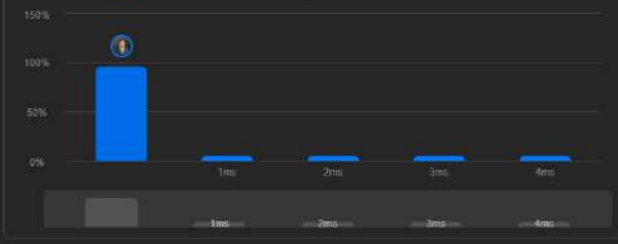
Runtime

0 ms | Beats 100.00%

Analyze Complexity

Memory

13.42 MB | Beats 39.75%



Code | C++

```
/**
 * Definition for singly-linked list.
```

Testcase | Test Result

Source

leetcode.com/problems/delete-the-middle-node-of-a-linked-list/submissions/1564889688/

YouTube | Maps | Outlook | Strivers A2Z DSA Sh... | My Courses | MDN Web Docs | Free Logo Icon | Ic... | OS Scaler | DBMS Scaler

Problem List | Run | Submit | Premium

Description | Editorial | Solutions | Submissions

2095. Delete the Middle Node of a Linked List

Solved

Medium | Topics | Companies | Hint

You are given the `head` of a linked list. **Delete** the **middle node**, and return *the* `head` *of the modified linked list.*

The **middle node** of a linked list of size `n` is the `⌊n / 2⌋th` node from the **start** using **0-based indexing**, where `⌊x⌋` denotes the largest integer less than or equal to `x`.

- For `n` = `1`, `2`, `3`, `4`, and `5`, the middle nodes are `0`, `1`, `1`, `2`, and `2`, respectively.

Example 1:


```
Input: head = [1,3,4,7,1,2,6]
Output: [1,3,4,1,2,6]
Explanation:
The above figure represents the given linked list. The indices of the nodes are written
below.
Since n = 7, node 3 with value 7 is the middle node, which is marked in red.
We return the new list after removing this node.
```

Example 2:

4.5K | 77 | 62 Online

Code | Accepted

All Submissions

Accepted 70 / 70 testcases passed

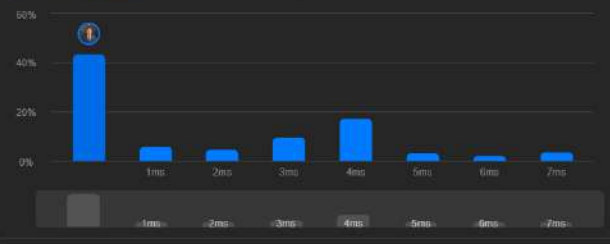
Mayank Khandelwal submitted at Mar 06, 2025 18:47

Editorial | Solution

Runtime 0 ms | Beats 100.00%

Analyze Complexity

Memory 312.19 MB | Beats 17.89%



Code | C++

Testcase | Test Result

```
Line 66: Char 28:
=================================================================
==22==ERROR: AddressSanitizer: heap-use-after-free on address 0x502000000bb8 at pc 0x55d4dbb89606 bp 0x7
ffd9b04da70 sp 0x7ffd9b04da60
READ of size 8 at 0x502000000bb8 thread T0
    #0 0x55d4dbb89605 in __ListNodeUtils__::hasCycle(ListNode*) (solution+0x1aa605)
```

leetcode.com/problems/merge-two-sorted-lists/description/

YouTube | Maps | Outlook | Strivers A2Z DSA Sh... | My Courses | MDN Web Docs | Free Logo Icon | Ic... | OS Scaler | DBMS Scaler

Problem List | Run | Submit | Premium

Description | Editorial | Solutions | Submissions

21. Merge Two Sorted Lists

Solved

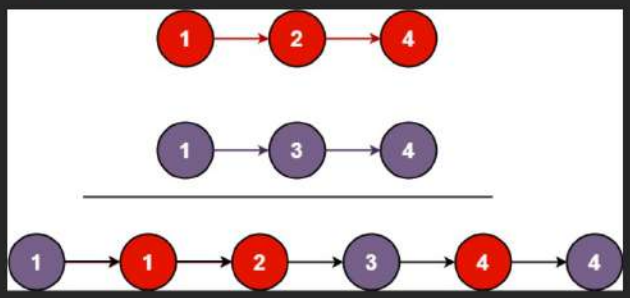
Easy | Topics | Companies

You are given the heads of two sorted linked lists `list1` and `list2`.

Merge the two lists into one **sorted** list. The list should be made by splicing together the nodes of the first two lists.

Return *the head of the merged linked list.*

Example 1:



Input: list1 = [1,2,4], list2 = [1,3,4]

23K | 420 | 404 Online

Code | Accepted

All Submissions

Accepted 208 / 208 testcases passed

Mayank Khandelwal submitted at Mar 06, 2025 17:02

Editorial | Solution

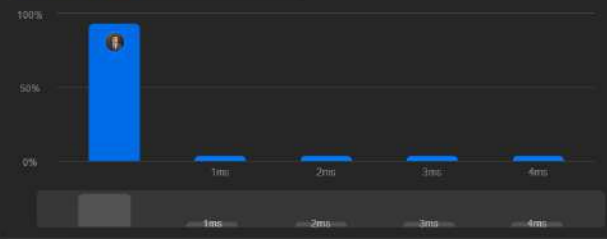
Runtime

0 ms | Beats 100.00%

Analyze Complexity

Memory

19.38 MB | Beats 86.51%



Code | C++

```
/**
 * Definition for singly-linked list.
 * struct ListNode {
 *     int val;
 *     ListNode *next;
```

Testcase | Test Result

leetcode.com/problems/linked-list-cycle/description/

YouTube | Maps | Outlook | Strivers A2Z DSA Sh... | My Courses | MDN Web Docs | Free Logo Icon | Ic... | OS Scaler | DBMS Scaler

Problem List | Run | Submit | Premium

Description | Editorial | Solutions | Submissions

141. Linked List Cycle

Solved

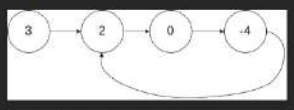
Easy | Topics | Companies

Given `head`, the head of a linked list, determine if the linked list has a cycle in it.

There is a cycle in a linked list if there is some node in the list that can be reached again by continuously following the `next` pointer. Internally, `pos` is used to denote the index of the node that tail's `next` pointer is connected to. **Note that `pos` is not passed as a parameter.**

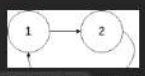
Return `true` *if there is a cycle in the linked list*. Otherwise, return `false`.

Example 1:




```
Input: head = [3,2,0,-4], pos = 1
Output: true
Explanation: There is a cycle in the linked list, where the tail connects to the 1st node (0-indexed).
```

Example 2:



16.2K | 360 | 224 Online

Code | Accepted

All Submissions

Accepted 29 / 29 testcases passed

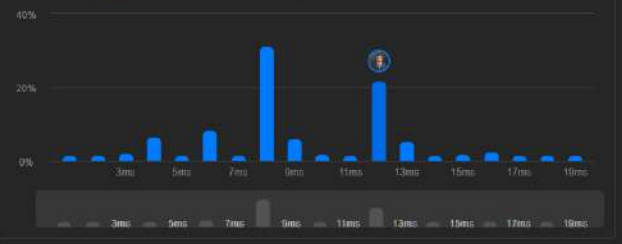
Mayank Khandelwal submitted at Mar 06, 2025 17:50

Editorial | Solution

Runtime 12 ms | Beats 40.41%

Analyze Complexity

Memory 11.86 MB | Beats 54.03%



Code | C++

```
/**
 * Definition for singly-linked list.
 * struct ListNode {
 *     int val;
 *     ListNode *next;
```

Testcase | Test Result

leetcode.com/problems/rotate-list/submissions/1564890839/

YouTube | Maps | Outlook | Strivers A2Z DSA Sh... | My Courses | MDN Web Docs | Free Logo Icon | Ic... | OS Scaler | DBMS Scaler

Problem List | Run | Submit | Premium

Description | Editorial | Solutions | Submissions

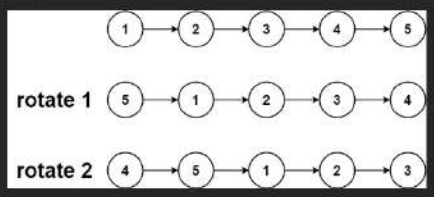
61. Rotate List

Solved

Medium | Topics | Companies

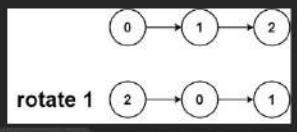
Given the `head` of a linked list, rotate the list to the right by `k` places.

Example 1:




```
Input: head = [1,2,3,4,5], k = 2
Output: [4,5,1,2,3]
```

Example 2:



10.2K | 104 | 92 Online

Accepted

All Submissions

Accepted 232 / 232 testcases passed

Mayank Khandelwal submitted at Mar 06, 2025 18:48

Editorial | Solution

Runtime

0 ms | Beats 100.00%

Analyze Complexity

Memory

16.43 MB | Beats 31.91%

Code

C++ | Auto

```
8  *     ListNode(int x, ListNode *next) : val(x), next(next) {}
9  * };
```

Saved | Ln 34, Col 3

Testcase | Test Result

Source

leetcode.com/problems/sort-list/description/

YouTube | Maps | Outlook | Strivers A2Z DSA Sh... | My Courses | MDN Web Docs | Free Logo Icon | Ic... | OS Scaler | DBMS Scaler

Problem List | Run | Submit | Premium

Description | Editorial | Solutions | Submissions

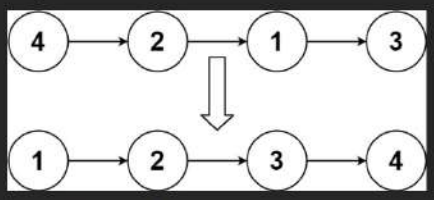
148. Sort List

Solved

Medium | Topics | Companies

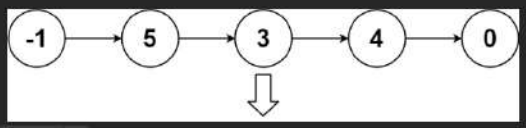
Given the `head` of a linked list, return *the list after sorting it in* ***ascending order***.

Example 1:



Input: head = [4,2,1,3]
Output: [1,2,3,4]

Example 2:



12.2K | 111 | 131 Online

Accepted

All Submissions

Accepted 30 / 30 testcases passed

Mayank Khandelwal submitted at Mar 06, 2025 18:49

Editorial | Solution

Runtime
11 ms | Beats 84.82%
Analyze Complexity

Memory
57.14 MB | Beats 81.38%

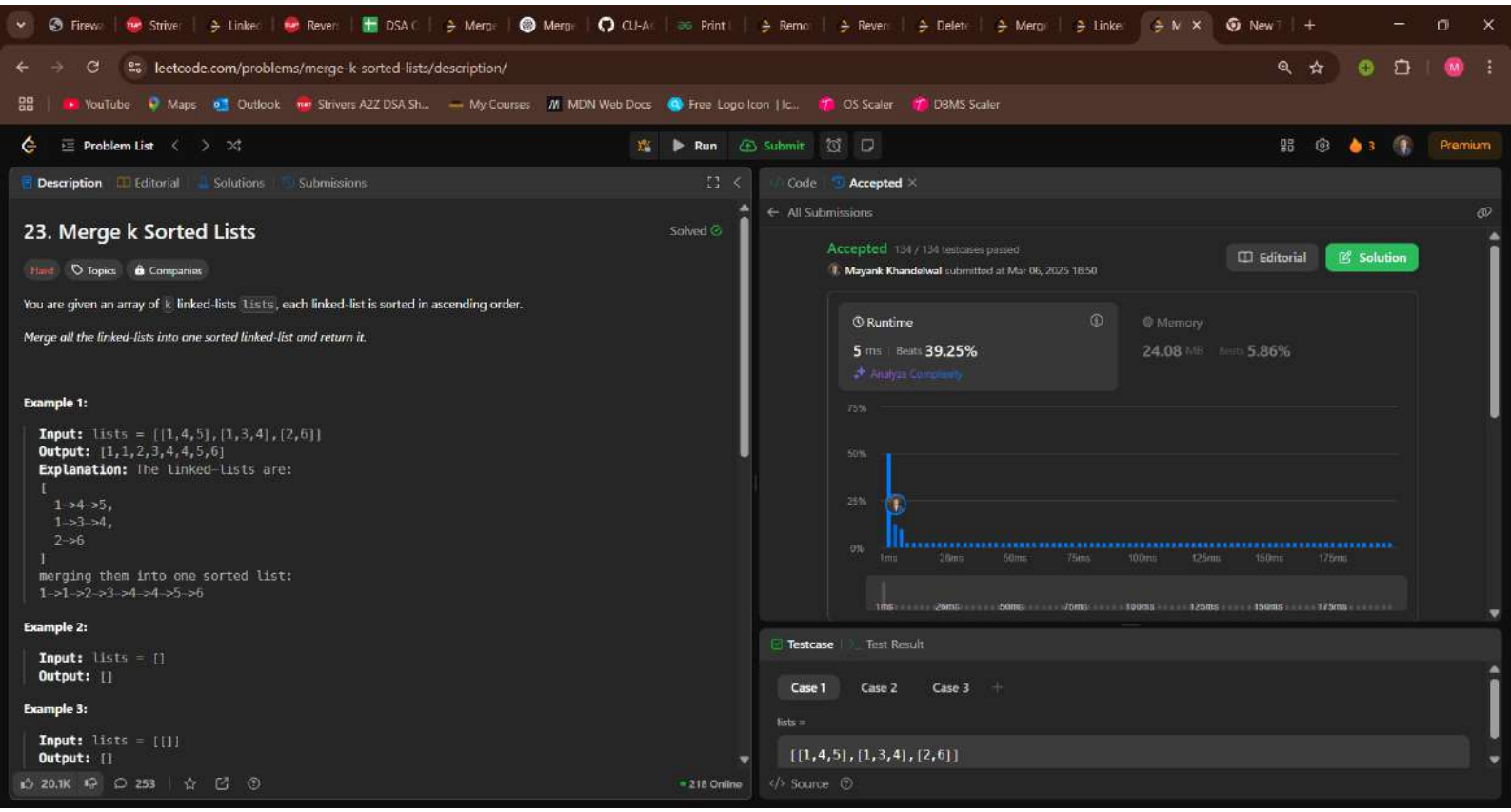
Code

C++ | Auto

```
16
17          ListNode* mid = slow->next;
18          slow->next = nullptr;
19
20          // Recursively split and merge
21          ListNode* left = sortList(head);
22          ListNode* right = sortList(mid);
23
24          return merge(left, right);
25      }
26
27      ListNode* merge(ListNode* l1, ListNode* l2) {
```

Saved | Ln 45, Col 3

Testcase | Test Result


leetcode.com/problems/merge-k-sorted-lists/description/
Problem List
Run
Submit
Premium
Description
Editorial
Solutions
Submissions
23. Merge k Sorted Lists
Solved
Hard
Topics
Companies
You are given an array of k linked-lists lists, each linked-list is sorted in ascending order.
Merge all the linked-lists into one sorted linked-list and return it.
Example 1:
Input: lists = [[1,4,5],[1,3,4],[2,6]]
Output: [1,1,2,3,4,4,5,6]
Explanation: The linked-lists are:
[
1->4->5,
1->3->4,
2->6
]
merging them into one sorted list:
1->1->2->3->4->4->5->6
Example 2:
Input: lists = []
Output: []
Example 3:
Input: lists = [[]]
Output: []
20.1K
253
218 Online
Code
Accepted
All Submissions
Accepted 134 / 134 testcases passed
Mayank Khandelwal submitted at Mar 06, 2025 18:50
Editorial
Solution
Runtime
5 ms | Beats 39.25%
Analyze Complexity
Memory
24.08 MB Beats 5.86%
75%
50%
25%
0%
1ms
25ms
50ms
75ms
100ms
125ms
150ms
175ms
Testcase
Test Result
Case 1
Case 2
Case 3
lists =
[[1,4,5],[1,3,4],[2,6]]
Source