

leetcode.com/problems/implement-queue-using-stacks/submissions/1578891016/

Problem List | Run | Submit | Premium

Description | Accepted | Editorial | Solutions | Submissions | Submit | [Edit](#) | [Copy](#)

All Submissions | [All Submissions](#) | [My Submissions](#) | [Recent Submissions](#)

Accepted 22 / 22 testcases passed ✓ submitted at Mar 19, 2025

Editorial | Solution | Sync w/ LeetHub

Runtime: 0 ms | Beats 100.00% Analyze Complexity

Memory: 9.61 MB | Beats 65.72%

Manually upload this submission to GitHub (beta). This will OVERWRITE your current submission. Please be mindful of your GitHub rate-limits.

```
#include <stack>
class MyQueue {
private:
    std::stack<int> s1, s2;
public:
    MyQueue() {}

    void push(int x) {
        s1.push(x);
    }

    int pop() {
        if (s2.empty()) {
            while (!s1.empty())
                s2.push(s1.top());
                s1.pop();
        }
        return s2.top();
    }

    int peek() {
        if (!s2.empty())
            return s2.top();
        else
            return s1.top();
    }

    bool empty() {
        return s1.empty() && s2.empty();
    }
};
```

Saved | Ln 1, Col 1

Testcase | Test Result

Accepted Runtime: 0 ms

Case 1

Input

```
["MyQueue","push","push","peek","pop","empty"]
```

```
[[],[1],[2],[],[],[]]
```

leetcode.com/problems/best-time-to-buy-and-sell-stock/submissions/1578900392/ ☆ ⌂ 🚙 📸

Problem List < > 🔍 Run Submit ⏱ ⏷ 0 Premium

Description | Accepted x | Editorial | Solutions | Submissions

All Submissions @

Accepted 212 / 212 testcases passed Sync w/ LeetHub

submitted at Mar 19, 2025

Runtime 0 ms | Beats 100.00% Analyze Complexity

Memory 97.44 MB | Beats 25.68%

75% 50% 25% 0% 1ms 2ms 3ms 4ms 5ms 6ms

Code C++ Auto

```
1 class Solution {
2 public:
3     int maxProfit(vector<int>& prices) {
4         int min_val = INT_MAX;
5         int max_profit = 0;
6
7         for (int i = 0; i < prices.size(); i++) {
8             min_val = min(min_val, prices[i]);
9             max_profit = max(max_profit, prices[i] - min_val);
10        }
11
12        return max_profit;
13    }
14 }
```

Saved Ln 1, Col 1

Testcase | >_ Test Result

Case 1 Case 2 +

prices = [7,1,5,3,6,4]

</> Source ?

leetcode.com/problems/maximum-subarray/submissions/1578900609/ ☆ ⌂ 🚙 📸 📜

G Problem List < > 🔍 Run Submit ⏱ ⌂ 0 Premium

Description | Accepted x | Editorial | Solutions | Submissions ⌂ <

All Submissions

Accepted 210 / 210 testcases passed ↻

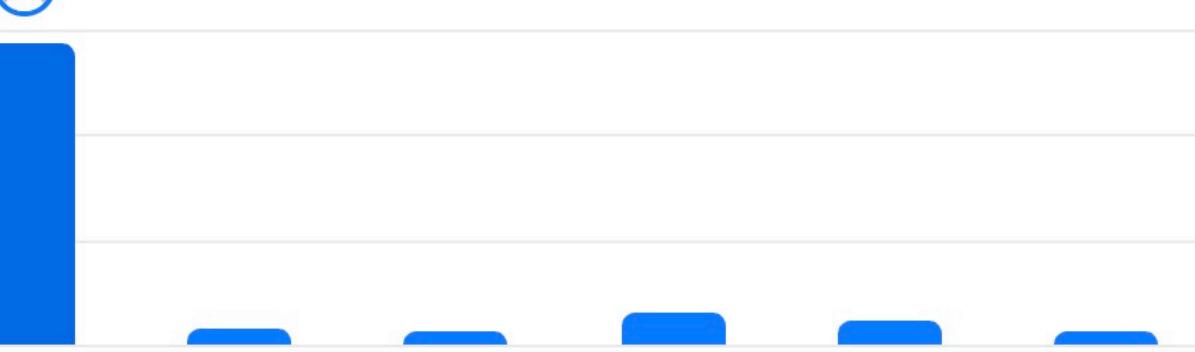
Editorial Solution Sync w/ LeetHub

submitted at Mar 19, 2025

Runtime
0 ms | Beats 100.00% ⚡

Analyze Complexity

Memory
71.70 MB | Beats 53.26% ⚡

100%  75% 50% 25% 0% 
1ms 2ms 3ms 4ms 5ms

Code

C++ Auto

```
1 class Solution {
2 public:
3     int maxSubArray(vector<int>& nums) {
4         int max_sum=INT_MIN;
5         int current_sum=0;
6         for(int i=0;i<nums.size();i++){
7             current_sum=current_sum+nums[i];
8             max_sum=max(current_sum,max_sum);
9             if(current_sum<0){
10                 current_sum=0;
11             }
12         }
13     }
14 }
```

Saved Ln 1, Col 1

Testcase | Test Result

Case 1 Case 2 Case 3 +

nums = [-2,1,-3,4,-1,2,1,-5,4]

Source ?

leetcode.com/problems/majority-element/submissions/1578900828/



 Problem List 

 Run Submit

   0  Premium

 Description |  Accepted   Editorial |  Solutions |  Submissions  <

[← All Submissions](#)

Accepted 52 / 52 testcases passed ✓

Editorial

Solution

 Sync w/ LeetHub

 submitted at Mar 19, 2025

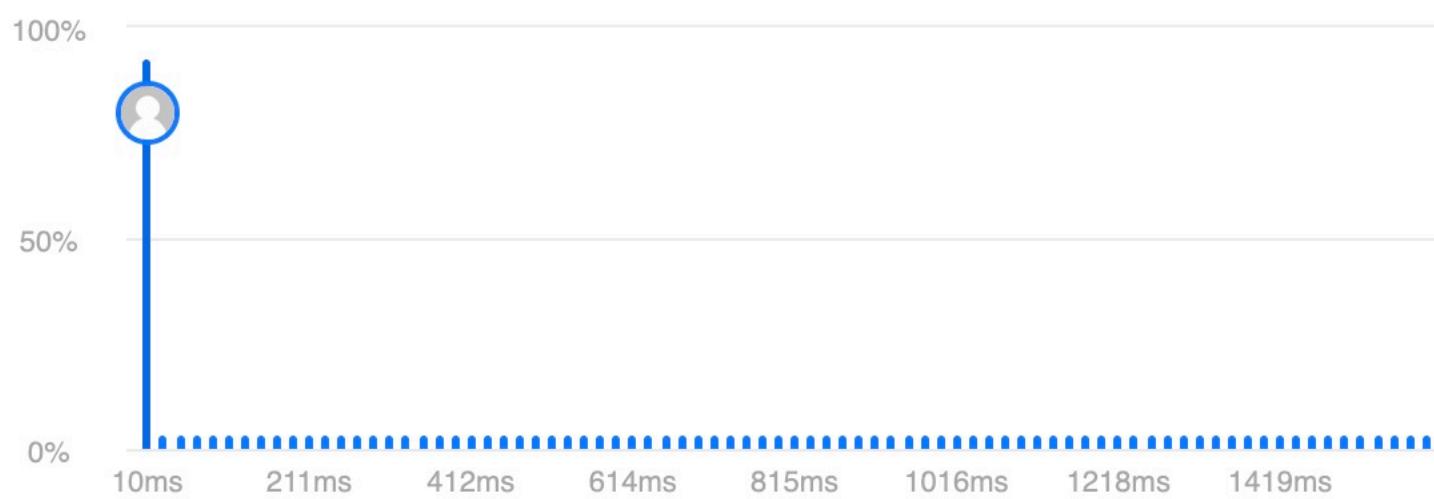
⌚ Runtime

0 ms | Beats 100.00% 

❖ Analyze Complexity

Memory

28.18 MB | Beats **58.72%**



Code

C++ <  Auto

```
1 class Solution {
2 public:
3     int majorityElement(vector<int>& nums) {
4         int n=nums.size();
5         int count=0;
6         int candidate=0;
7         for(int i=0;i<n;i++){
8             if(count==0){
9                 candidate=nums[i];
10                count=1;
11            }
12            else if(nums[i]==candidate){
13                count++;
14            }
15        }
16        return candidate;
17    }
18 }
```

Saved

Ln 1, Col 1

 Testcase |  Test Result

Case 1

Case 2

+

nums =

[3,2,3]

</> Source ?

leetcode.com/problems/sort-colors/submissions/1578901101/ star copy profile more

Problem List < > refresh Run Submit lock refresh 0 Premium

Description Accepted Editorial Solutions Submissions

All Submissions at

Accepted 89 / 89 testcases passed refresh

Editorial Solution Sync w/ LeetHub

submitted at Mar 19, 2025

Runtime 0 ms | Beats 100.00% analyze complexity

Memory 11.67 MB | Beats 38.99%

150%
100%
50%
0%
1ms 2ms 3ms 4ms

Code C++ Auto

```
1 class Solution {  
2     public:  
3  
4         void sortColors(vector<int>& nums) {  
5             for(int i=0;i<nums.size();i++){  
6                 for(int j=i+1;j<nums.size();j++){  
7                     if(nums[i]>nums[j]){  
8                         swap(nums[i],nums[j]);  
9                     }  
10                }  
11            }  
12        }  
13    }  
14}
```

Saved Ln 1, Col 1

Testcase Test Result

Case 1 Case 2 +

nums = [2,0,2,1,1,0]

Source ?

leetcode.com/problems/two-sum/submissions/1578901326/

Problem List | Run | Submit | Premium

Description | Accepted | Editorial | Solutions | Submissions | All Submissions | @

Accepted 63 / 63 testcases passed ✓

Editorial Solution Sync w/ LeetHub

submitted at Mar 19, 2025

Runtime
43 ms | Beats 30.99% | Analyze Complexity

Memory
14.15 MB | Beats 61.19% 🌟

Manually upload this submission to GitHub (beta). This will OVERWRITE your current submission. Please be mindful of your GitHub rate-limits.

Code

C++ Auto

```
1 class Solution {
2 public:
3     vector<int> twoSum(vector<int>& nums, int target) {
4         for(int i=0;i<nums.size();i++){
5             for(int j=i+1;j<nums.size();j++){
6                 if(nums[i]+nums[j]==target){
7                     return {i,j};
8                 }
9             }
10        }
11    }
12 }
13
14 }
```

Saved Ln 1, Col 1

Testcase | Test Result

Case 1 Case 2 Case 3 +

nums = [2, 7, 11, 15]

target =

</> Source ?

18ms 36ms 53ms 71ms 89ms 106ms 124ms

[Problem List](#) < >

Run Submit

0 Premium

[Description](#) | [Editorial](#) | [Solutions](#) | [Submissions](#)

<

5. Longest Palindromic Substring

Solved

[Medium](#) [Topics](#) [Companies](#) [Hint](#)

Given a string `s`, return the longest *palindromic substring* in `s`.

Example 1:

Input: `s = "babad"`

Output: "bab"

Explanation: "aba" is also a valid answer.

Example 2:

Input: `s = "cbbd"`

Output: "bb"

Constraints:

- `1 <= s.length <= 1000`
- `s` consist of only digits and English letters.

</> Code

C++ v Auto

```
1 class Solution {
2 public:
3     string longestPalindrome(string s) {
4         if(s.empty()) return "";
5         int start=0,maxLength=1;
6
7         auto expandAroundCentre=[&](int left,int right){
8             while(left>=0 && right<s.length() && s[left]==s[right]){
9                 left--;
10                right++;
11            }
12            return right-left-1;
13        };
14        for(int i=0;i<s.length();i++){
```

Restored from local [Upgrade to Cloud Saving](#)

Ln 1, Col 1

 Testcase | [Test Result](#)[Case 1](#) [Case 2](#) +

s =

`"babad"`

leetcode.com/problems/longest-palindromic-substring/submissions/1578901840/ star copy profile

Problem List < > refresh Run Submit Copy 0 Premium

Description Accepted Editorial Solutions Submissions

All Submissions edit

Accepted 142 / 142 testcases passed Editorial Solution Sync w/ LeetHub

submitted at Mar 19, 2025

Runtime 7 ms | Beats 88.63% Analyze Complexity

Memory 9.48 MB | Beats 74.33% Profile

60%

Code C++ Auto

```
1 class Solution {
2 public:
3     string longestPalindrome(string s) {
4         if(s.empty()) return "";
5         int start=0,maxLength=1;
6
7         auto expandAroundCentre=[&](int left,int right){
8             while(left>=0 && right<s.length() && s[left]==s[right]){
9                 left--;
10                right++;
11            }
12            return right-left-1;
13        };
14        for(int i=0;i<s.length();i++){

```

Saved Ln 1, Col 1

Testcase Test Result

Case 1 Case 2 +

s = "babad"

Source ?

leetcode.com/problems/count-number-of-homogenous-substrings/submissions/1578902099/ star copy profile

Problem List < > refresh Run Submit Copy 0 Premium

Description | Accepted x | Editorial | Solutions | Submissions

All Submissions at

Accepted 85 / 85 testcases passed refresh

Editorial Solution Sync w/ LeetHub

submitted at Mar 19, 2025

Runtime
4 ms | Beats 59.89% analyze

Analyze Complexity

Memory
14.90 MB | Beats 75.62% analyze

Performance Histogram (ms): 5ms (~15%), 10ms (~6%), 10ms (~6%), 5ms (~5%), 10ms (~5%), 5ms (~5%), 10ms (~5%), 28ms (~3%), 10ms (~2%), 28ms (~2%), 44ms (~1%), 51ms (~1%)

Code

C++ Auto

```
1 class Solution {
2 public:
3     int countHomogenous(string s) {
4
5         long long result=0;
6         long long count=1;
7         const int MOD=1e9+7;
8
9         for(int i=1;i<s.size();i++){
10             if(s[i]==s[i-1]){
11                 count++;
12             }else{
13                 result=(result+(count*(count+1)/2))%MOD;
14                 count=1;
15             }
16         }
17         return result;
18     }
19 }
```

Saved Ln 1, Col 1

Testcase | Test Result

Case 1 Case 2 Case 3 +

s = "abbcccaa"

Source ?

leetcode.com/problems/string-to-integer-atoi/submissions/1578902303/ star copy profile

Problem List < > refresh Run Submit Copy 0 Premium

Description | Accepted x | Editorial | Solutions | Submissions

All Submissions at

Accepted 1095 / 1095 testcases passed refresh Editorial Solution Sync w/ LeetHub

submitted at Mar 19, 2025

Runtime 0 ms | Beats 100.00% Analyze Complexity

Memory 9.10 MB | Beats 57.46% graph

100%  75% 50% 25% 0% 1ms 2ms 3ms 4ms

Code C++ Auto

```
1 class Solution {
2 public:
3     int myAtoi(string s) {
4         int i=0;
5         int n=s.size();
6         while(i<n && s[i]==' ')
7             i++;
8
9
10        int sign = 1;
11        if (i < n && (s[i] == '+' || s[i] == '-')) {
12            char firstSign = s[i];
13            i++;
14        }
15
16        int result = 0;
17        while(i<n && s[i]>='0' && s[i]<='9') {
18            result = result * 10 + s[i] - '0';
19            i++;
20        }
21
22        if (sign == -1)
23            result = -result;
24
25        return result;
26    }
27}
```

Testcase Test Result

Case 1 Case 2 Case 3 Case 4 Case 5 +

s = "42"

Source ?

leetcode.com/problems/roman-to-integer/submissions/1578902710/ star copy profile

Problem List < > refresh Run Submit lock timer close grid gear flame 0 Premium

Description Accepted Editorial Solutions Submissions all submissions

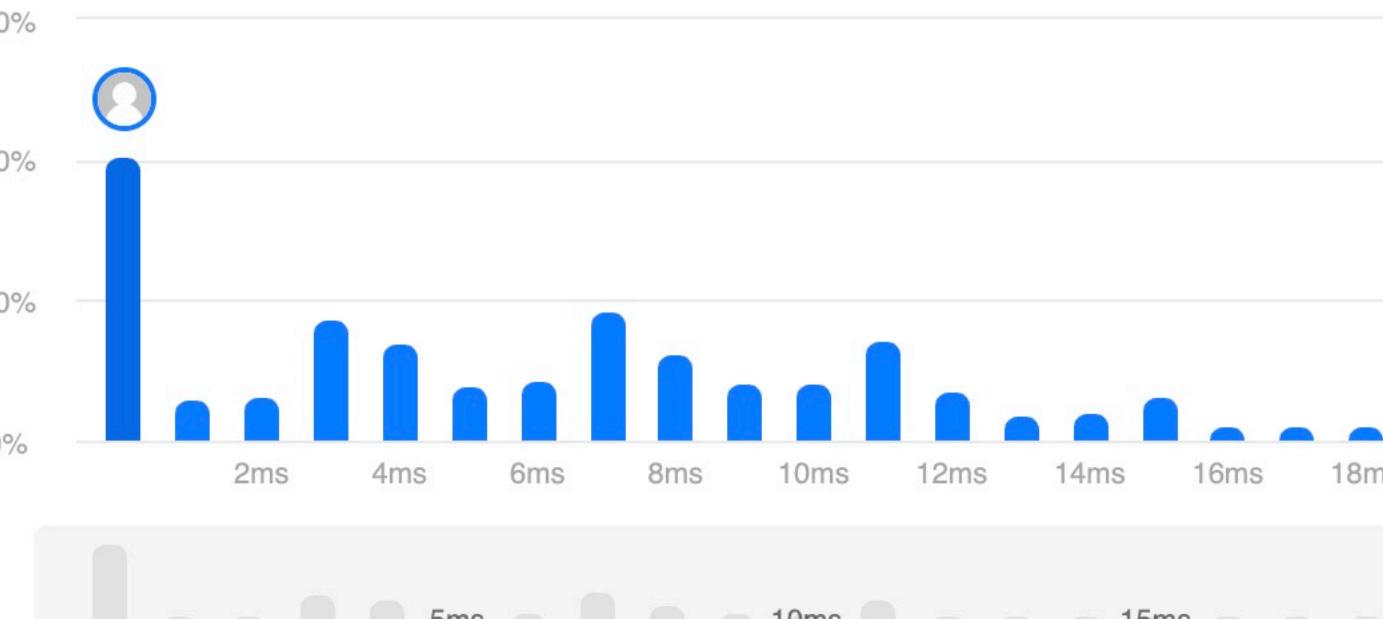
All Submissions link

Accepted 3999 / 3999 testcases passed refresh Editorial Solution Sync w/ LeetHub

submitted at Mar 19, 2025

Runtime 0 ms | Beats 100.00% Analyze Complexity

Memory 13.42 MB | Beats 19.26%

30%  20%
10%
0% 2ms 4ms 6ms 8ms 10ms 12ms 14ms 16ms 18ms 5ms 10ms 15ms

Code C++ Auto

```
1 class Solution {
2 public:
3     int romanToInt(string s) {
4         unordered_map<char,int> roman={
5             {'I',1},{'V',5},{'X',10},{'L',50},{'C',100}
6             ,{'D',500},{'M',1000}
7         };
8         int result=0;
9         int prevval=0;
10        for(int i=s.size()-1;i>=0;i--){
11            int currval=roman[s[i]];
12            if(currval<prevval){
13                result-=currval;
14            }else {
```

Saved Ln 1, Col 1

Testcase Test Result

Case 1 Case 2 Case 3 +

s = "III"

Source ?

leetcode.com/problems/linked-list-cycle-ii/submissions/1578902953/ star copy profile

Problem List < > refresh Run Submit Copy 0 Premium

Description | Accepted x | Editorial | Solutions | Submissions ... <

All Submissions ... @

Accepted 18 / 18 testcases passed refresh Editorial Solution Sync w/ LeetHub

submitted at Mar 19, 2025

Runtime 7 ms | Beats 68.08% analyze

Memory 11.22 MB | Beats 83.00% analyze

Manually upload this submission to GitHub (beta). This will OVERWRITE your current submission. Please be mindful of your GitHub rate-limits.

Code C++ Auto

```
1 /**
2  * Definition for singly-linked list.
3  * struct ListNode {
4  *     int val;
5  *     ListNode *next;
6  *     ListNode(int x) : val(x), next(NULL) {}
7  * };
8 */
9 class Solution {
10 public:
11     ListNode *detectCycle(ListNode *head) {
12         ListNode *slow = head, *fast = head;
13         while (fast && fast->next) {
14             slow = slow->next;
```

Saved Ln 1, Col 1

Testcase Test Result

Case 1 Case 2 Case 3 +

head = [3, 2, 0, -4]

pos =

</> Source ?

Time Range (ms)	Percentage (%)
0-2	~1%
2-4	~1%
4-6	~1%
6-8	~1%
8-10	~1%
10-12	~1%
12-14	~1%
14-16	~1%
16-18	~1%
18-20	~1%
20-22	~1%
22-24	~1%
24-26	~1%
26-28	~1%
28-30	~1%
30-32	~1%
32-34	~1%
34-36	~1%
36-38	~1%
38-40	~1%
40-42	~1%
42-44	~1%
44-46	~1%
46-48	~1%
48-50	~1%
50-52	~1%
52-54	~1%
54-56	~1%
56-58	~1%
58-60	~1%
60-62	~1%
62-64	~1%
64-66	~1%
66-68	~1%
68-70	~1%
70-72	~1%
72-74	~1%
74-76	~1%
76-78	~1%
78-80	~1%
80-82	~1%
82-84	~1%
84-86	~1%
86-88	~1%
88-90	~1%
90-92	~1%
92-94	~1%
94-96	~1%
96-98	~1%
98-100	~1%
100-102	~1%
102-104	~1%
104-106	~1%
106-108	~1%
108-110	~1%
110-112	~1%
112-114	~1%
114-116	~1%
116-118	~1%
118-120	~1%
120-122	~1%
122-124	~1%
124-126	~1%
126-128	~1%
128-130	~1%
130-132	~1%
132-134	~1%
134-136	~1%
136-138	~1%
138-140	~1%
140-142	~1%
142-144	~1%
144-146	~1%
146-148	~1%
148-150	~1%
150-152	~1%
152-154	~1%
154-156	~1%
156-158	~1%
158-160	~1%
160-162	~1%
162-164	~1%
164-166	~1%
166-168	~1%
168-170	~1%
170-172	~1%
172-174	~1%
174-176	~1%
176-178	~1%
178-180	~1%
180-182	~1%
182-184	~1%
184-186	~1%
186-188	~1%
188-190	~1%
190-192	~1%
192-194	~1%
194-196	~1%
196-198	~1%
198-200	~1%
200-202	~1%
202-204	~1%
204-206	~1%
206-208	~1%
208-210	~1%
210-212	~1%
212-214	~1%
214-216	~1%
216-218	~1%
218-220	~1%
220-222	~1%
222-224	~1%
224-226	~1%
226-228	~1%
228-230	~1%
230-232	~1%
232-234	~1%
234-236	~1%
236-238	~1%
238-240	~1%
240-242	~1%
242-244	~1%
244-246	~1%
246-248	~1%
248-250	~1%
250-252	~1%
252-254	~1%
254-256	~1%
256-258	~1%
258-260	~1%
260-262	~1%
262-264	~1%
264-266	~1%
266-268	~1%
268-270	~1%
270-272	~1%
272-274	~1%
274-276	~1%
276-278	~1%
278-280	~1%
280-282	~1%
282-284	~1%
284-286	~1%
286-288	~1%
288-290	~1%
290-292	~1%
292-294	~1%
294-296	~1%
296-298	~1%
298-300	~1%
300-302	~1%
302-304	~1%
304-306	~1%
306-308	~1%
308-310	~1%
310-312	~1%
312-314	~1%
314-316	~1%
316-318	~1%
318-320	~1%
320-322	~1%
322-324	~1%
324-326	~1%
326-328	~1%
328-330	~1%
330-332	~1%
332-334	~1%
334-336	~1%
336-338	~1%
338-340	~1%
340-342	~1%
342-344	~1%
344-346	~1%
346-348	~1%
348-350	~1%
350-352	~1%
352-354	~1%
354-356	~1%
356-358	~1%
358-360	~1%
360-362	~1%
362-364	~1%
364-366	~1%
366-368	~1%
368-370	~1%
370-372	~1%
372-374	~1%
374-376	~1%
376-378	~1%
378-380	~1%
380-382	~1%
382-384	~1%
384-386	~1%
386-388	~1%
388-390	~1%
390-392	~1%
392-394	~1%
394-396	~1%
396-398	~1%
398-400	~1%
400-402	~1%
402-404	~1%
404-406	~1%
406-408	~1%
408-410	~1%
410-412	~1%
412-414	~1%
414-416	~1%
416-418	~1%
418-420	~1%
420-422	~1%
422-424	~1%
424-426	~1%
426-428	~1%
428-430	~1%
430-432	~1%
432-434	~1%
434-436	~1%
436-438	~1%
438-440	~1%
440-442	~1%
442-444	~1%
444-446	~1%
446-448	~1%
448-450	~1%
450-452	~1%
452-454	~1%
454-456	~1%
456-458	~1%
458-460	~1%
460-462	~1%
462-464	~1%
464-466	~1%
466-468	~1%
468-470	~1%
470-472	~1%
472-474	~1%
474-476	~1%
476-478	~1%
478-480	~1%
480-482	~1%
482-484	~1%
484-486	~1%
486-488	~1%
488-490	~1%
490-492	~1%
492-494	~1%
494-496	~1%
496-498	~1%
498-500	~1%
500-502	~1%
502-504	~1%
504-506	~1%
506-508	~1%
508-510	~1%
510-512	~1%
512-514	~1%
514-516	~1%
516-518	~1%
518-520	~1%
520-522	~1%
522-524	~1%
524-526	~1%
526-528	~1%
528-530	~1%
530-532	~1%
532-534	~1%
534-536	~1%
536-538	~1%
538-540	~1%
540-542	~1%
542-544	~1%
544-546	~1%
546-548	~1%
548-550	~1%
550-552	~1%
552-554	~1%
554-556	~1%
556-558	~1%
558-560	~1%
560-562	~1%
562-564	~1%
564-566	~1%
566-568	~1%
568-570	~1%
570-572	~1%
572-574	~1%
574-576	~1%
576-578	~1%
578-580	~1%
580-582	~1%
582-584	~1%
584-586	~1%
586-588	~1%
588-590	~1%
590-592	~1%
592-594	~1%
594-596	~1%
596-598	~1%
598-600	~1%
600-602	~1%
602-604	~1%
604-606	~1%
606-608	~1%
608-610	~1%
610-612	~1%
612-614	~1%
614-616	~1%
616-618	~1%

leetcode.com/problems/sort-list/submissions/1578903184/

Problem List < > 🔍 Run Submit ⏱ 0 Premium

Description | Accepted x | Editorial | Solutions | Submissions

All Submissions

Accepted 30 / 30 testcases passed submitted at Mar 19, 2025

Editorial Solution Sync w/ LeetHub

Runtime 14 ms | Beats 71.89%

Analyze Complexity

Memory 55.79 MB | Beats 99.18%

Testcase | Test Result

Case 1 Case 2 Case 3 +

head = [4, 2, 1, 3]

Source ?

```
1 /**
2  * Definition for singly-linked list.
3  * struct ListNode {
4  *     int val;
5  *     ListNode *next;
6  *     ListNode() : val(0), next(nullptr) {}
7  *     ListNode(int x) : val(x), next(nullptr) {}
8  *     ListNode(int x, ListNode *next) : val(x), next(next) {}
9  * };
10 */
11 class Solution {
12 public:
13     ListNode* sortList(ListNode* head) {
14         if(!head || !head->next) return head;
```

leetcode.com/problems/rotate-list/submissions/1578903504/

Problem List | Run | Submit | Premium

Description | Accepted | Editorial | Solutions | Submissions | All Submissions | @

Accepted | 232 / 232 testcases passed | submitted at Mar 19, 2025

Editorial | Solution | Sync w/ LeetHub

Manually upload this submission to GitHub (beta). This will OVERWRITE your current submission. Please be mindful of your GitHub rate-limits.

Runtime | 0 ms | Beats 100.00% | Analyze Complexity

Memory | 16.51 MB | Beats 31.82%

100% | 50% | 0% | 1ms | 2ms | 3ms | 4ms | 1ms | 2ms | 3ms | 4ms

Code

C++ | Auto

```
1 /**
2 * Definition for singly-linked list.
3 * struct ListNode {
4 *     int val;
5 *     ListNode *next;
6 *     ListNode() : val(0), next(nullptr) {}
7 *     ListNode(int x) : val(x), next(nullptr) {}
8 *     ListNode(int x, ListNode *next) : val(x), next(next) {}
9 * };
10 */
11 class Solution {
12 public:
13     ListNode* rotateRight(ListNode* head, int k) {
14         if (!head) return nullptr;
```

Saved | Ln 1, Col 1

Testcase | Test Result

Case 1 | Case 2 | +

head = [1,2,3,4,5]

k =

Source | ?

leetcode.com/problems/remove-duplicates-from-sorted-list-ii/submissions/1578903729/ star copy profile

Problem List < > refresh Run Submit lock refresh 0 Premium

Description Accepted Editorial Solutions Submissions

All Submissions edit

Accepted 166 / 166 testcases passed refresh

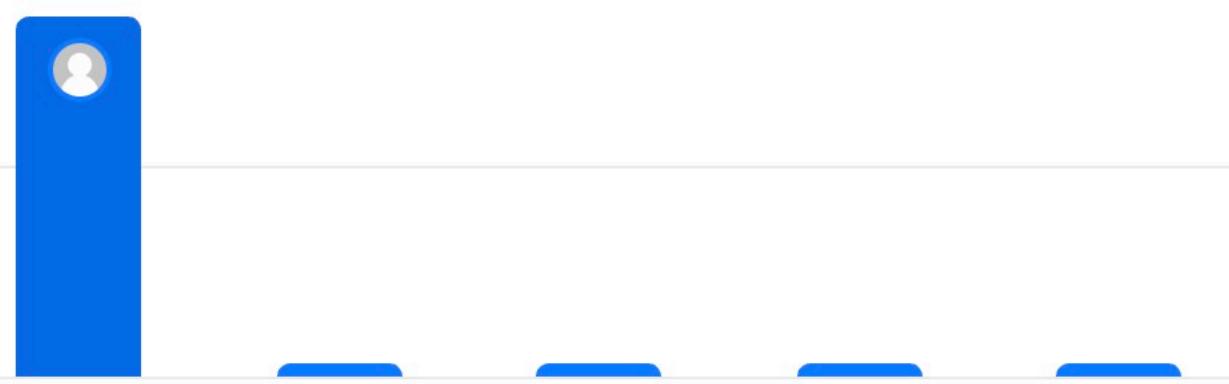
Editorial Solution Sync w/ LeetHub

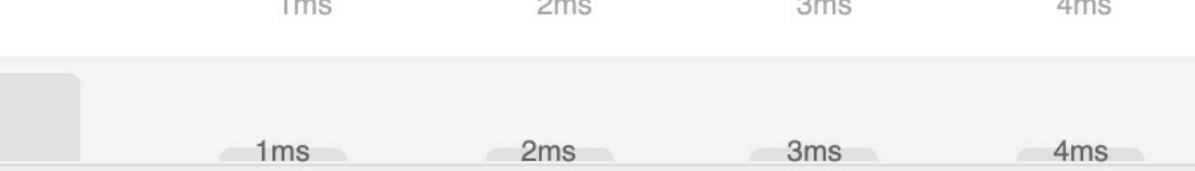
submitted at Mar 19, 2025

Runtime 0 ms | Beats 100.00% analyze

Analyze Complexity

Memory 15.68 MB | Beats 72.16% analyze

100% 
50%
0%

1ms 2ms 3ms 4ms 

Code C++ Auto

```
13     ListNode* deleteDuplicates(ListNode* head) {  
14         ListNode dummy(0, head);  
15         ListNode* prev = &dummy;  
16         while (head) {  
17             if (head->next && head->val == head->next->val)  
18                 head = head->next;  
19             if (prev->next == head)  
20                 prev = prev->next;  
21             else  
22                 prev->next = head->next;  
23             head = head->next;  
24         }  
25         return dummy.next;  
26     }
```

Saved Ln 1, Col 1

Testcase Test Result

Case 1 Case 2 +

head = [1,2,3,3,4,4,5]

Source ?