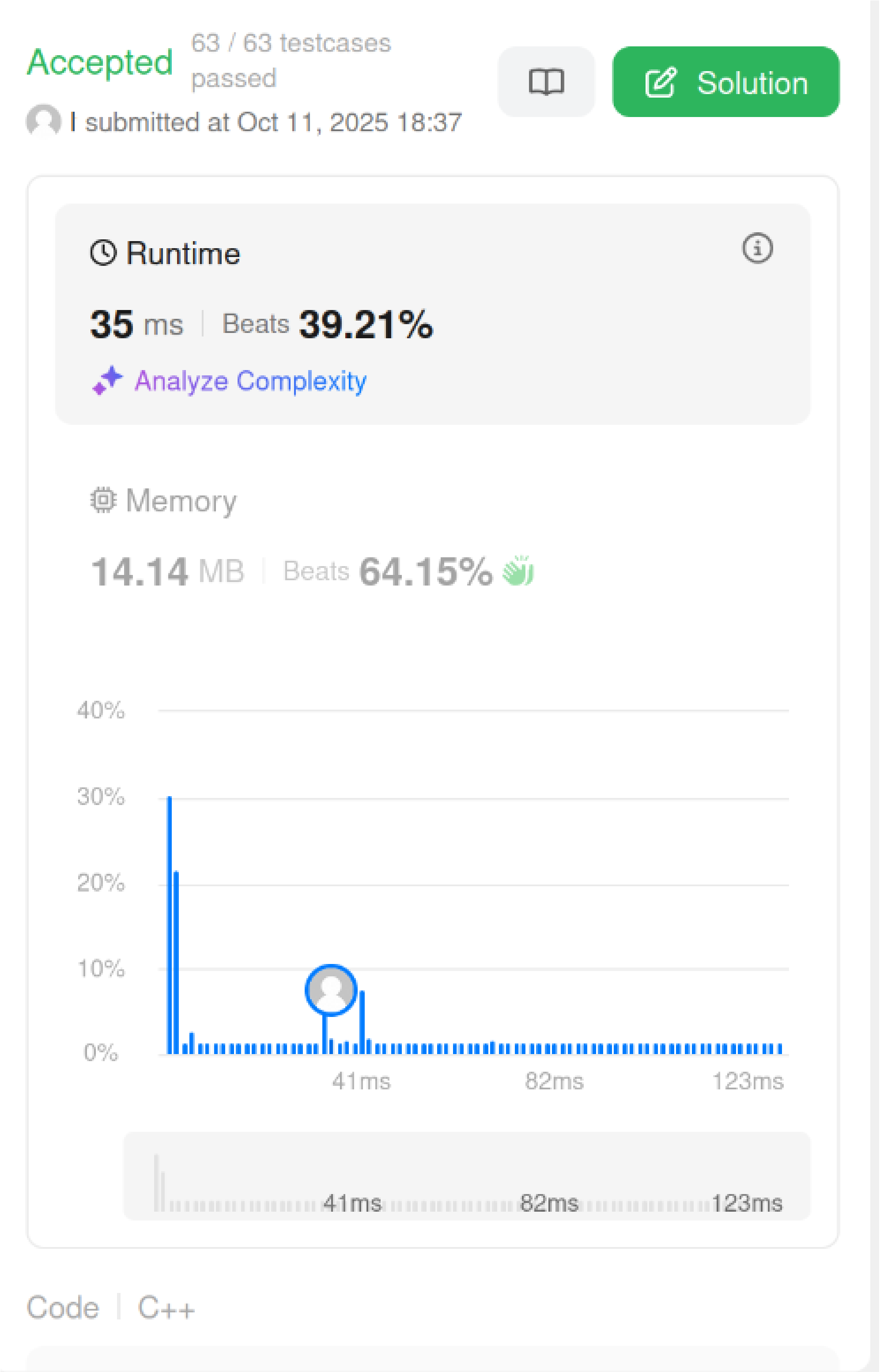
第一周练习

张睿恒 2024302182001

T1 LeetCode1 简单

```cpp
class Solution {
public:
    vector<int> twoSum(vector<int>& nums, int target) {
        int n = nums.size();
        for(int i=0;i<n;++i){
            for(int j=i+1;j<n;++j){
                if(nums[i] + nums[j] == target) return {i,j};
            }
        }
        return {};
    }

};
```



Accepted 63 / 63 testcases passed

I submitted at Oct 11, 2025 18:37

Runtime

35 ms | Beats 39.21%

Memory

14.14 MB | Beats 64.15%

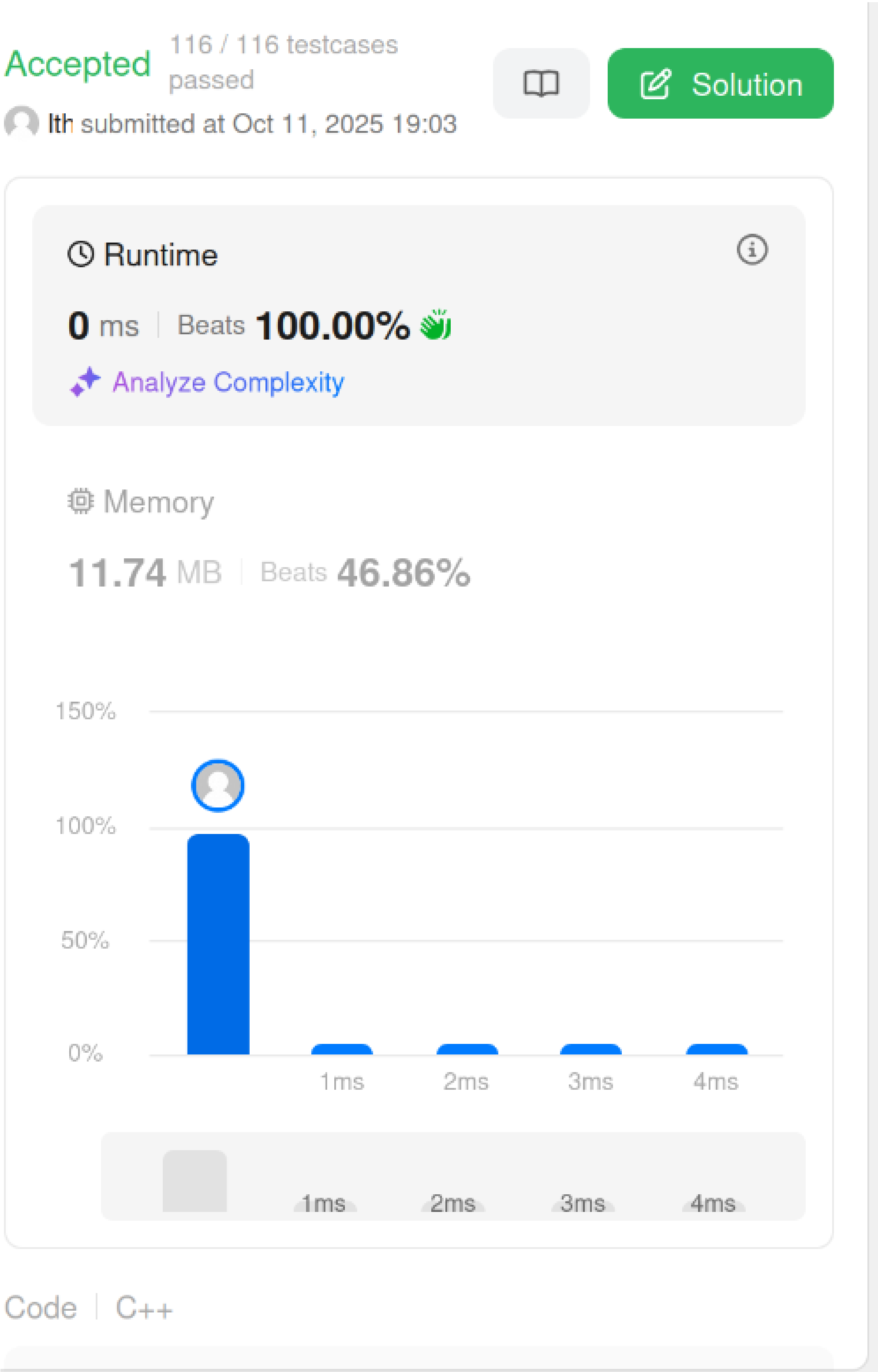
Code | C++

T2 LeetCode27 简单

```
class Solution {
 public:
  int removeElement(vector<int>& nums, int val) {
    int i = 0;

    for (const int num : nums)
      if (num != val)
        nums[i++] = num;

    return i;
  }
};
```



Accepted 116 / 116 testcases passed

lth submitted at Oct 11, 2025 19:03

Solution

Runtime

0 ms | Beats 100.00%

Analyze Complexity

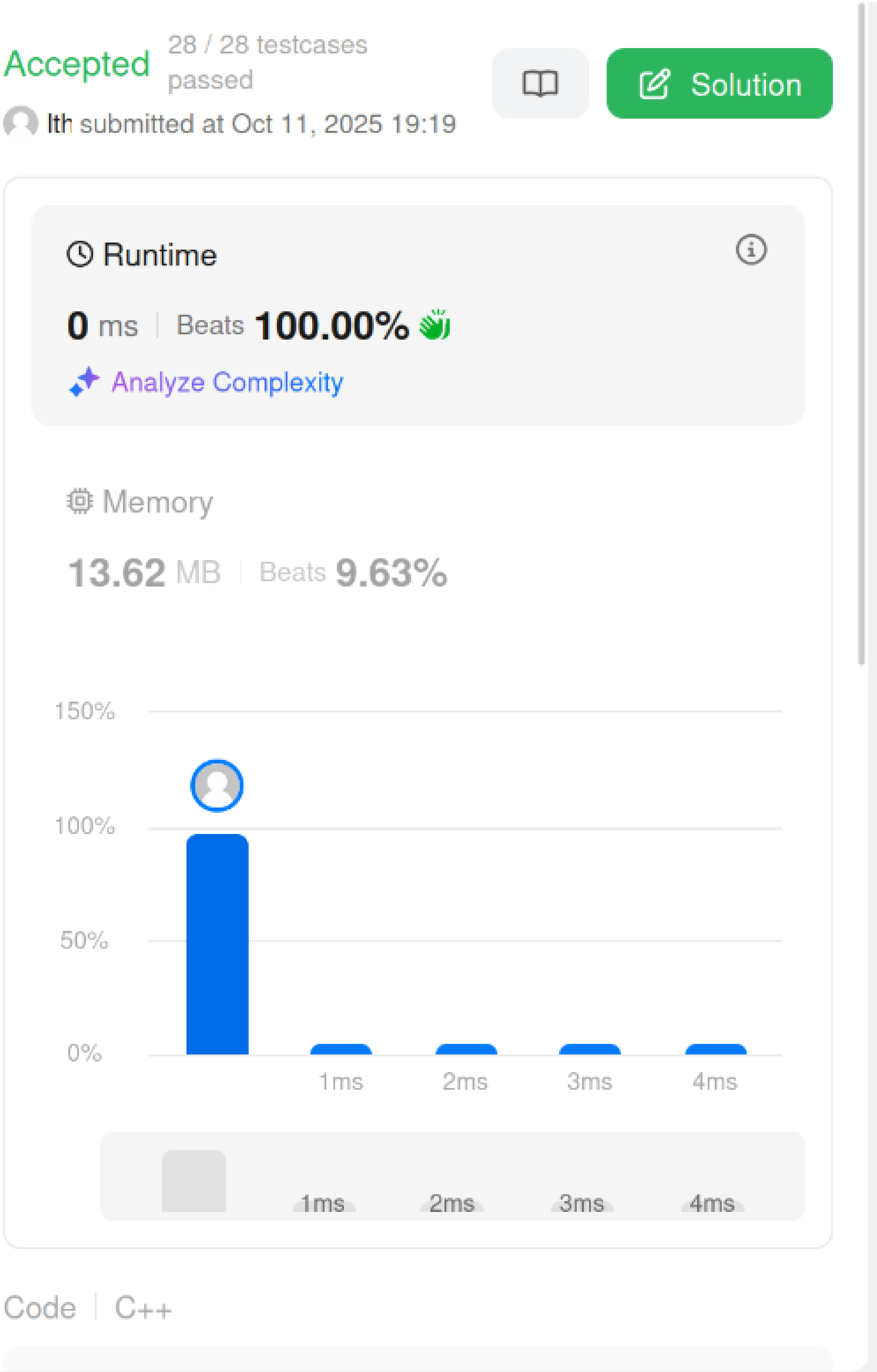
Memory

11.74 MB | Beats 46.86%

Code | C++

T3 LeetCode206 中等

```
/**
 * Definition for singly-linked list.
 * struct ListNode {
 *     int val;
 *     ListNode *next;
 *     ListNode() : val(0), next(nullptr) {}
 *     ListNode(int x) : val(x), next(nullptr) {}
 *     ListNode(int x, ListNode *next) : val(x), next(next) {}
 * };
 */
class Solution {
public:
    ListNode* reverseList(ListNode* head) {
        if(head == NULL or head->next == NULL) return head;
        ListNode* newHead = reverseList(head->next);
        head->next->next = head;
        head->next = NULL;
        return newHead;
    }
};
```



Accepted 28 / 28 testcases passed

lth submitted at Oct 11, 2025 19:19

Runtime

0 ms | Beats 100.00%

Analyze Complexity

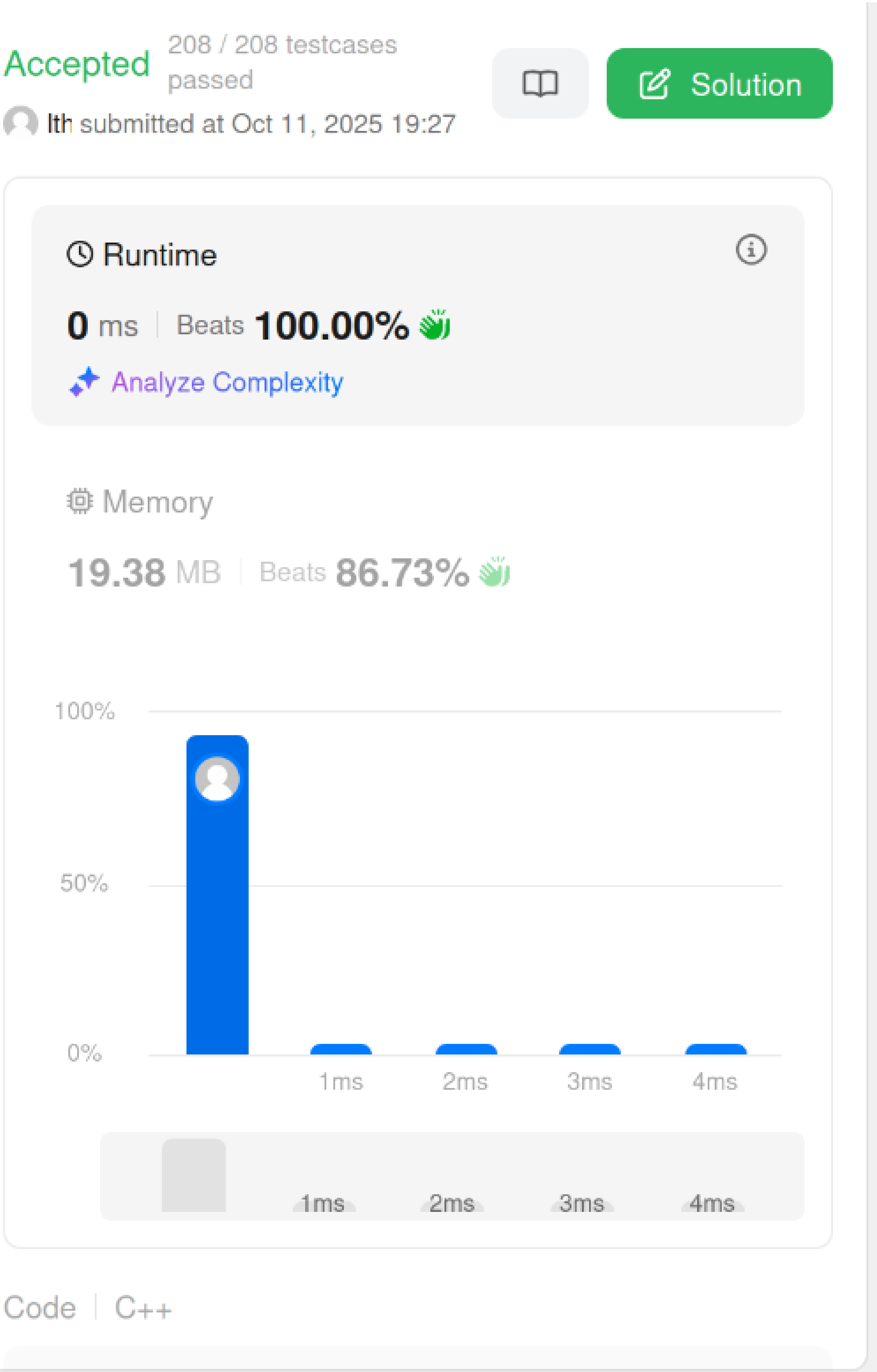
Memory

13.62 MB | Beats 9.63%

Code | C++

T5 LeetCode21 中等

```
/**
 * Definition for singly-linked list.
 * struct ListNode {
 *     int val;
 *     ListNode *next;
 *     ListNode() : val(0), next(nullptr) {}
 *     ListNode(int x) : val(x), next(nullptr) {}
 *     ListNode(int x, ListNode *next) : val(x), next(next) {}
 * };
 */
class Solution {
public:
    ListNode* mergeTwoLists(ListNode* list1, ListNode* list2) {
        if(list1 == NULL or list2 == NULL) return list1 ? list1 : list2;
        if(list1->val > list2->val) swap(list1 , list2);
        list1->next = mergeTwoLists(list1->next , list2);
        return list1;
    }
};
```



Accepted 208 / 208 testcases passed

lth submitted at Oct 11, 2025 19:27

Solution

Runtime

0 ms | Beats 100.00%

Analyze Complexity

Memory

19.38 MB | Beats 86.73%

Code | C++

T7 LeetCode42 困难

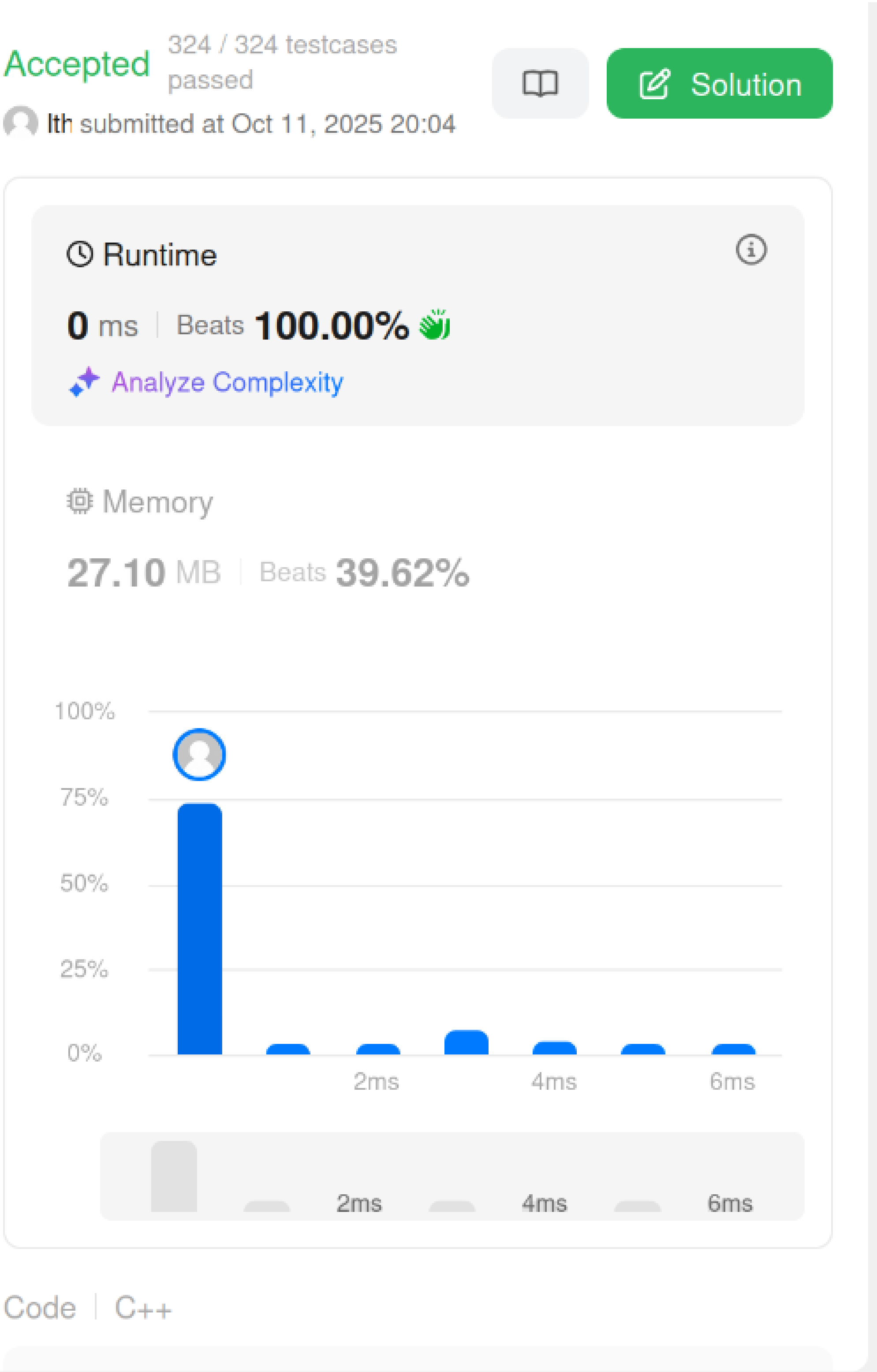
```
class Solution {
 public:
  int trap(vector<int>& height) {
    const int n = height.size();
    int ans = 0;
    vector<int> l(n);
    vector<int> r(n);

    l[0] = height[0];
    r[n-1] = height[n-1];
    for (int i = 1; i < n; ++i)
      l[i] = max(height[i], l[i - 1]);

    for (int i = n - 2; i >= 0; --i)
      r[i] = max(height[i], r[i + 1]);

    for (int i = 0; i < n; ++i)
      ans += min(l[i], r[i]) - height[i];

    return ans;
  }
};
```



Accepted 324 / 324 testcases passed

lth submitted at Oct 11, 2025 20:04

Solution

Runtime

0 ms | Beats 100.00%

Analyze Complexity

Memory

27.10 MB | Beats 39.62%

Code | C++

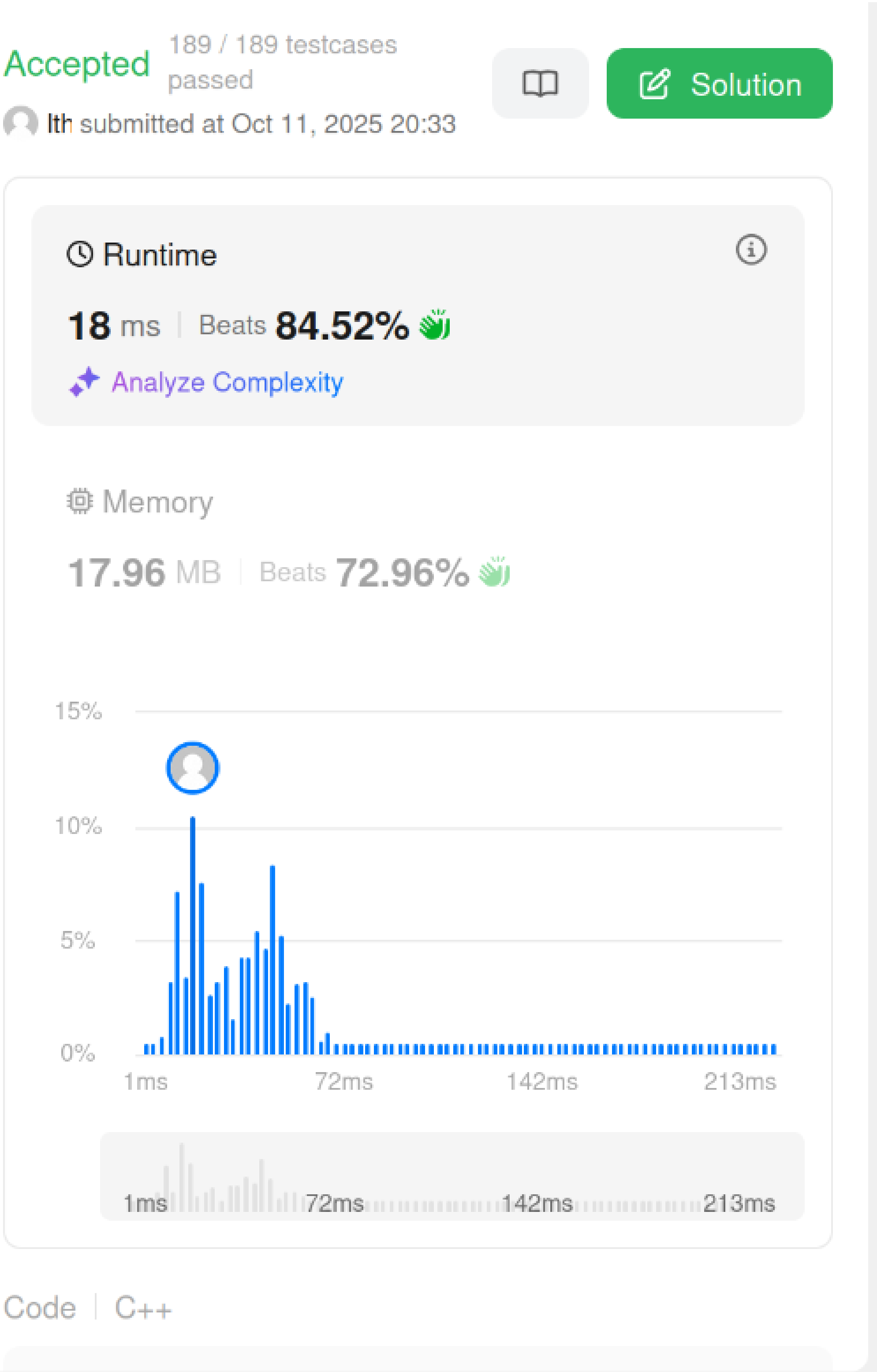
T8 LeetCode322 困难

```
class Solution {
 public:
  int coinChange(vector<int>& coins, int amount) {

    vector<int> dp(amount + 1, amount + 1);
    dp[0] = 0;

    for (int i=1;i<=amount;++i)
      for (const int coin : coins)
        if (coin <= i) dp[i] = min(dp[i],dp[i-coin]+1);

    if(dp[amount] == amount+1) return -1;
    return dp[amount];
  }
};
```



Accepted 189 / 189 testcases passed

lth submitted at Oct 11, 2025 20:33

Solution

Runtime

18 ms | Beats 84.52%

Analyze Complexity

Memory

17.96 MB | Beats 72.96%

Code | C++