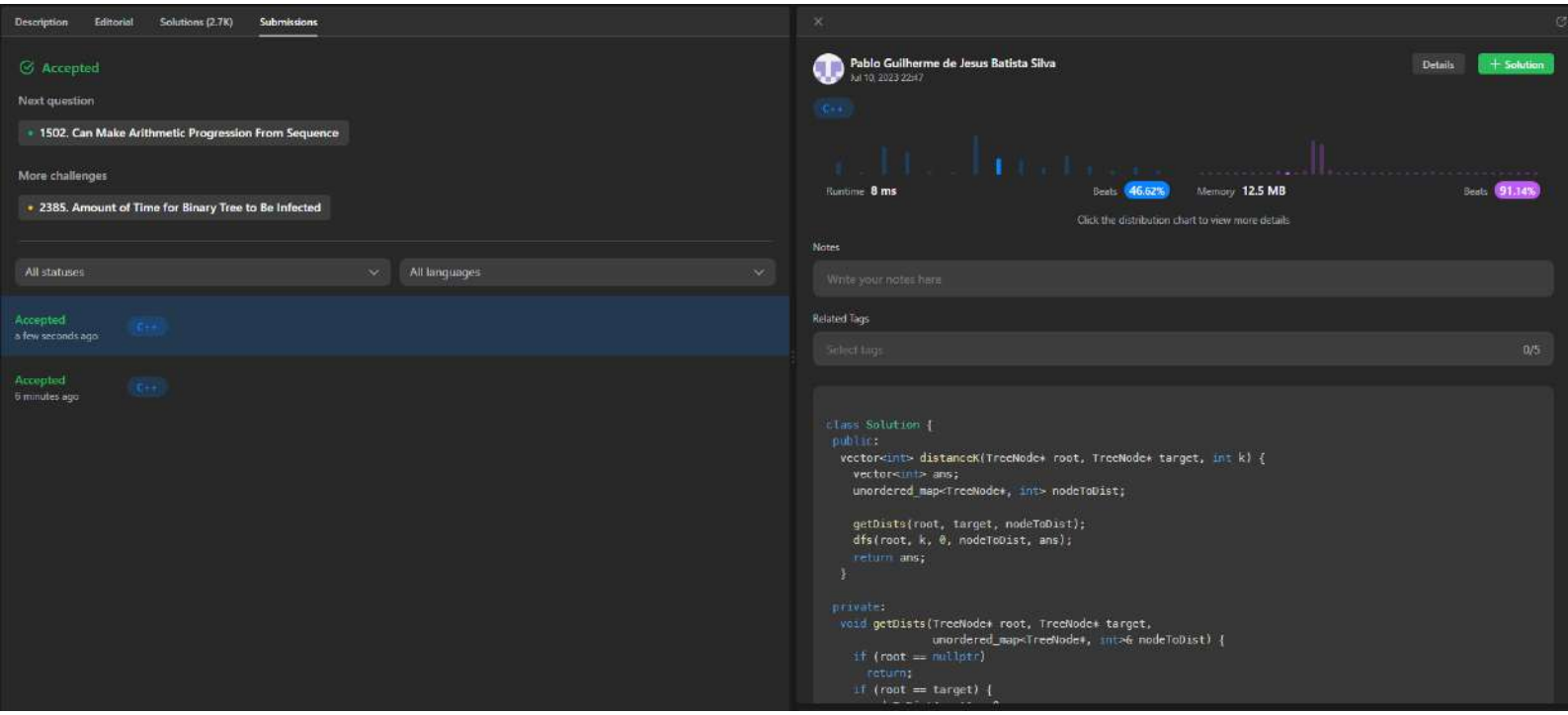
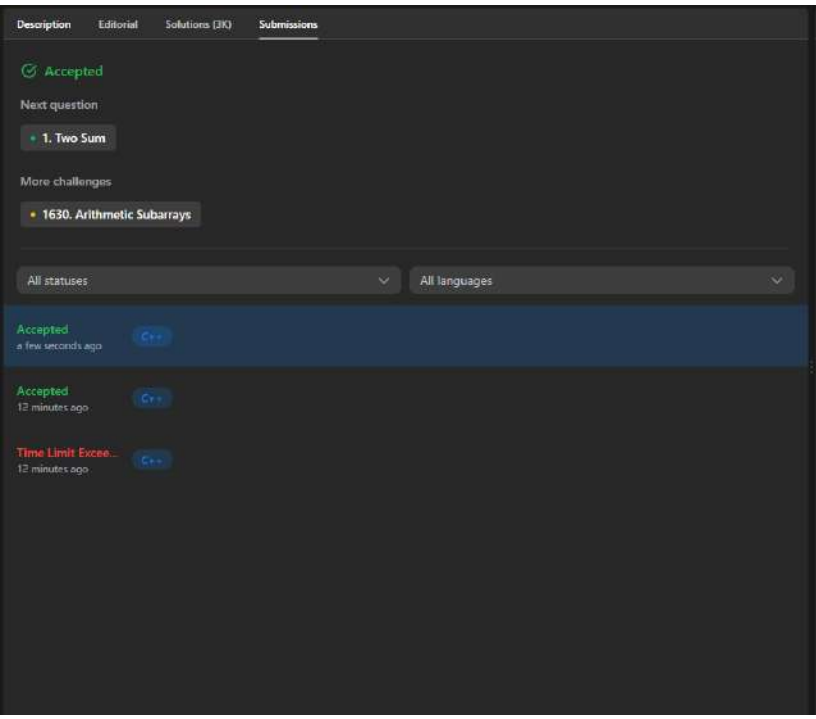
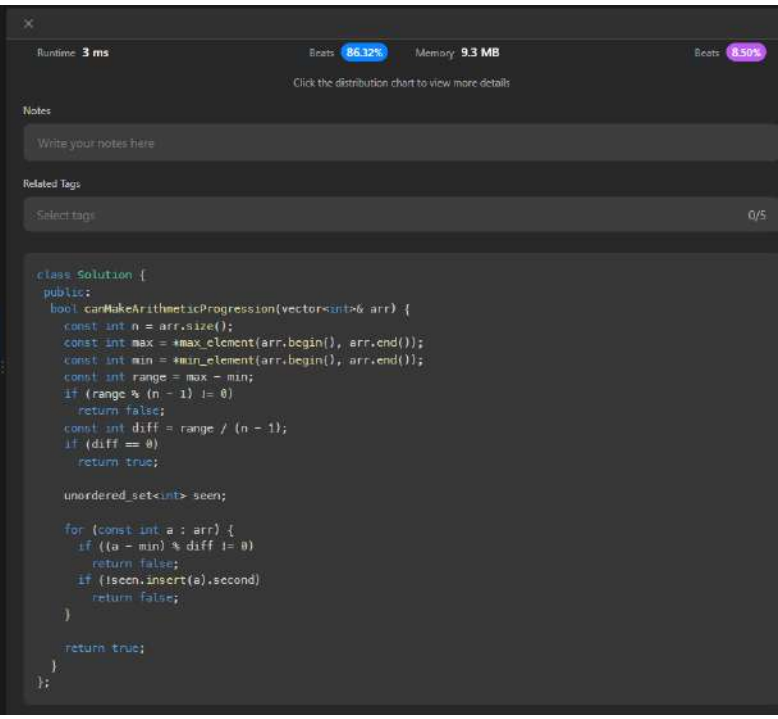

Description
Editorial
Solutions (2.7K)
Submissions
Accepted
Next question
1502. Can Make Arithmetic Progression From Sequence
More challenges
2385. Amount of Time for Binary Tree to Be Infected
All statuses
All languages
Accepted
a few seconds ago
C++
Accepted
6 minutes ago
C++
Pablo Guilherme de Jesus Batista Silva
Jul 10, 2023 22:47
Details
Solution
C++
Runtime 8 ms
Beats 46.62%
Memory 12.5 MB
Beats 91.14%
Click the distribution chart to view more details
Notes
Write your notes here
Related Tags
Select tags
0/5
class Solution {
public:
vector<int> distanceK(TreeNode* root, TreeNode* target, int k) {
vector<int> ans;
unordered_map<TreeNode*, int> nodeToDist;
getDists(root, target, nodeToDist);
dfs(root, k, 0, nodeToDist, ans);
return ans;
}
private:
void getDists(TreeNode* root, TreeNode* target,
unordered_map<TreeNode*, int>& nodeToDist) {
if (root == nullptr)
return;
if (root == target) {






```cpp
class Solution {
 public:
  bool canMakeArithmeticProgression(vector<int>& arr) {
    const int n = arr.size();
    const int max = *max_element(arr.begin(), arr.end());
    const int min = *min_element(arr.begin(), arr.end());
    const int range = max - min;
    if (range % (n - 1) != 0)
      return false;
    const int diff = range / (n - 1);
    if (diff == 0)
      return true;

    unordered_set<int> seen;

    for (const int a : arr) {
      if ((a - min) % diff != 0)
        return false;
      if (!seen.insert(a).second)
        return false;
    }

    return true;
  }
};
```