k-diff Pairs in an Array (LC-532)

2) 3 1 4 1 5 (0 1 2 3 4) $k = 2$

(1) Brute force :-

→ Consider each & every pair = & diff.

```
→ for (i=0; i<n; i++)
    for (j=i+1; j<n; j++)
      { if (abs(A[i]-A[j]) == k)
          { count++; }
      }
```

K-diff Pairs in an Array (LC-532)

2) 3 1 4 1 5 $K = 2$

0 1 2 3 4

① Brute force :-

→ Consider each & every pair & diff.

→
```
for (i=0; i<n-1; i++)
    for (j=i+1; j<n; j++)
        if (abs(A[i]-A[j]) == K)
            Count++;
```

$O(n^2)$ ≡

(2) Two pointer approach

K = 2

1	1	3	4	5
0	1	2	3	4
i	j			

abs(3 − 4) ⇒ 1 < 2

(1)
```
diff = a[j] - a[i]
if (diff = k)
{
    ans =
    i++ ; j++;
}
else if (diff > k)
{
    i++;
}
else
{
    j
}
```

(2) Two pointer approach

k = 2

1 | 1 | 3 | 4 | 5

0 1 2 3 4

i j

abs(3 − 4) ⇒ 1 < 2

① 1 | 1 | 3 | 4 | 5

i j

① diff ⇒ 0 (diff < k)

j++.

1 | 1 | 3 | 4 | 5

① diff = a[i] − a[j]

if (diff = k)

{ ans =

i++; j++;

}

else if (diff > k)

{ i++;

}

else

{ j

}

(2) Two pointer approach

k = 2

1 | 1 | 3 | 4 | 5

0 1 2 3 4

i j

abs(3 − 4) ⇒ 1 < 2

(1) 1 | 1 | 3 | 4 | 5

i j

(1) diff ⇒ 0 (diff < k)

j++.

1 | 1 | 3 | 4 | 5

i j

(1) diff = a[i] − a[j]

if (diff = k)

{ ans =

i++; j++;

}

else if (diff > k)

{ i++;

}

else

{ j

}

1 | 1 | 3 | 4 | 5

i j

(2) diff ⇒ 3 - 1 = (2)

2 = k

1, 3?

1 | 1 | 3 | 4 | 5

i i i j

(2) diff ⇒ 3 − 1 = (2)

2 = k

(1, 3) ~

(3) diff ⇒ 4 − 1 = 3

3 > 2

(4) diff ⇒ 4 − 3 = 1

(1 ≯ 2)

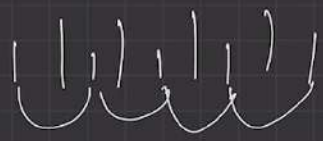
(5) diff = (5 − 3) ⇒ 2

(2 = k)

(3, 5) ~

Case → 1|3|1|5|4

→ 1|1|3|5|4 K=0.

2) 

K<=0.

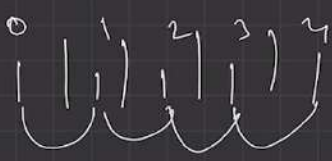
2) 1 1 ①
1 1 ②
1 1 ③
1 1 ④

Case → 1|3|1|5|4

→ 1|1|3|5|4

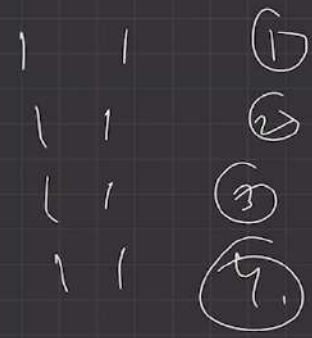
K = 0.

2)

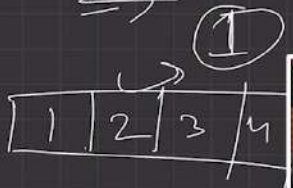


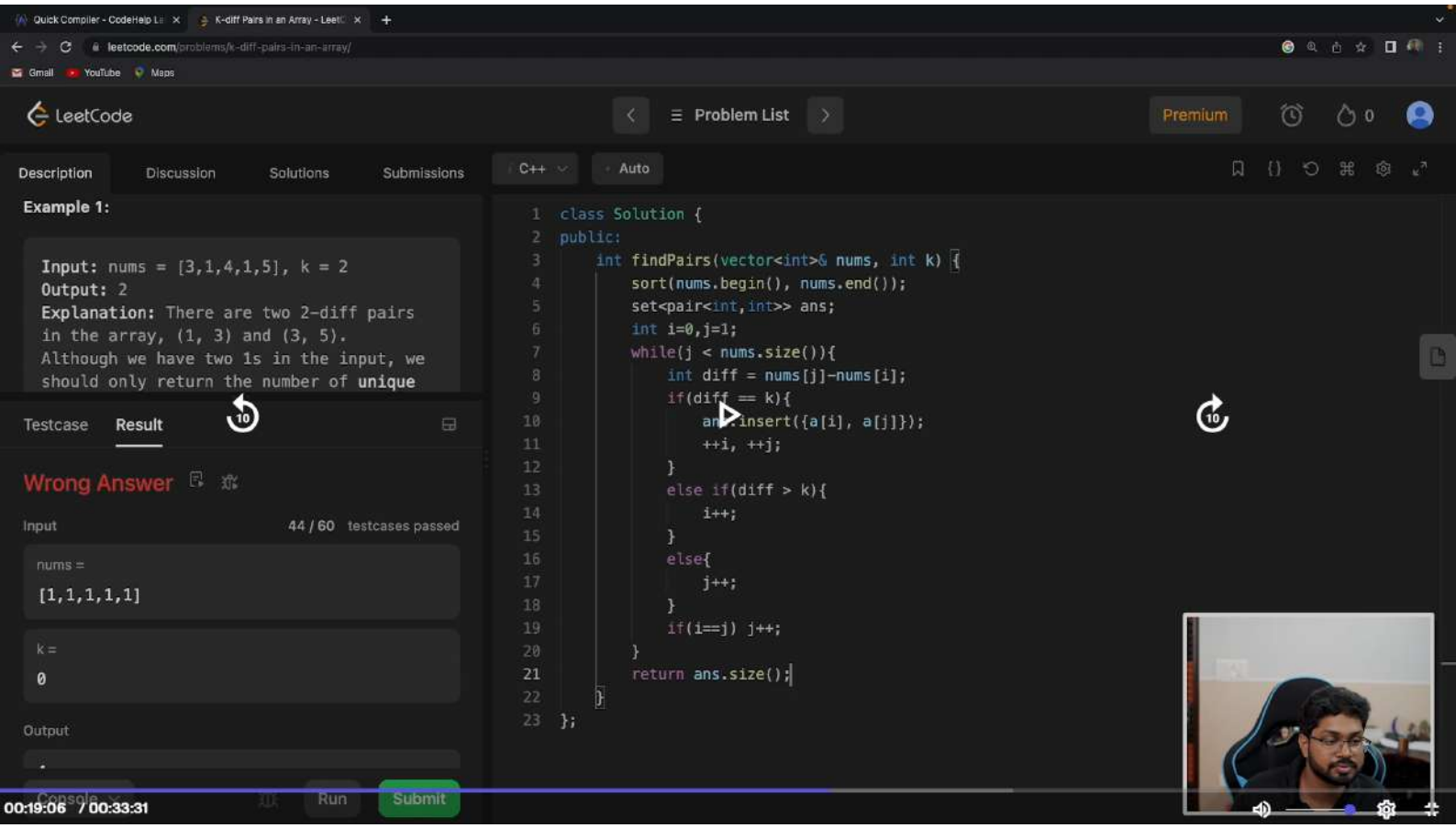
K <= 0.

2)



stl: set → int




Quick Compiler - CodeHelp La
K-diff Pairs in an Array - LeetC
leetcode.com/problems/k-diff-pairs-in-an-array/
Gmail
YouTube
Maps
LeetCode
Problem List
Premium
0
Description
Discussion
Solutions
Submissions
C++
Auto
Example 1:
Input: nums = [3,1,4,1,5], k = 2
Output: 2
Explanation: There are two 2-diff pairs in the array, (1, 3) and (3, 5).
Although we have two 1s in the input, we should only return the number of unique
Testcase
Result
Wrong Answer
Input
44 / 60 testcases passed
nums =
[1,1,1,1,1]
k =
0
Output
Run
Submit
00:19:06 / 00:33:31
class Solution {
public:
int findPairs(vector<int>& nums, int k) {
sort(nums.begin(), nums.end());
set<pair<int,int>> ans;
int i=0,j=1;
while(j < nums.size()){
int diff = nums[j]-nums[i];
if(diff == k){
ans.insert({a[i], a[j]});
++i, ++j;
}
else if(diff > k){
i++;
}
else{
j++;
}
if(i==j) j++;
}
return ans.size();
}
};

Next Method

→ Sorted

k = 2

⇒ 1 | 1 | 3 | 4 | 5 → Sorted,

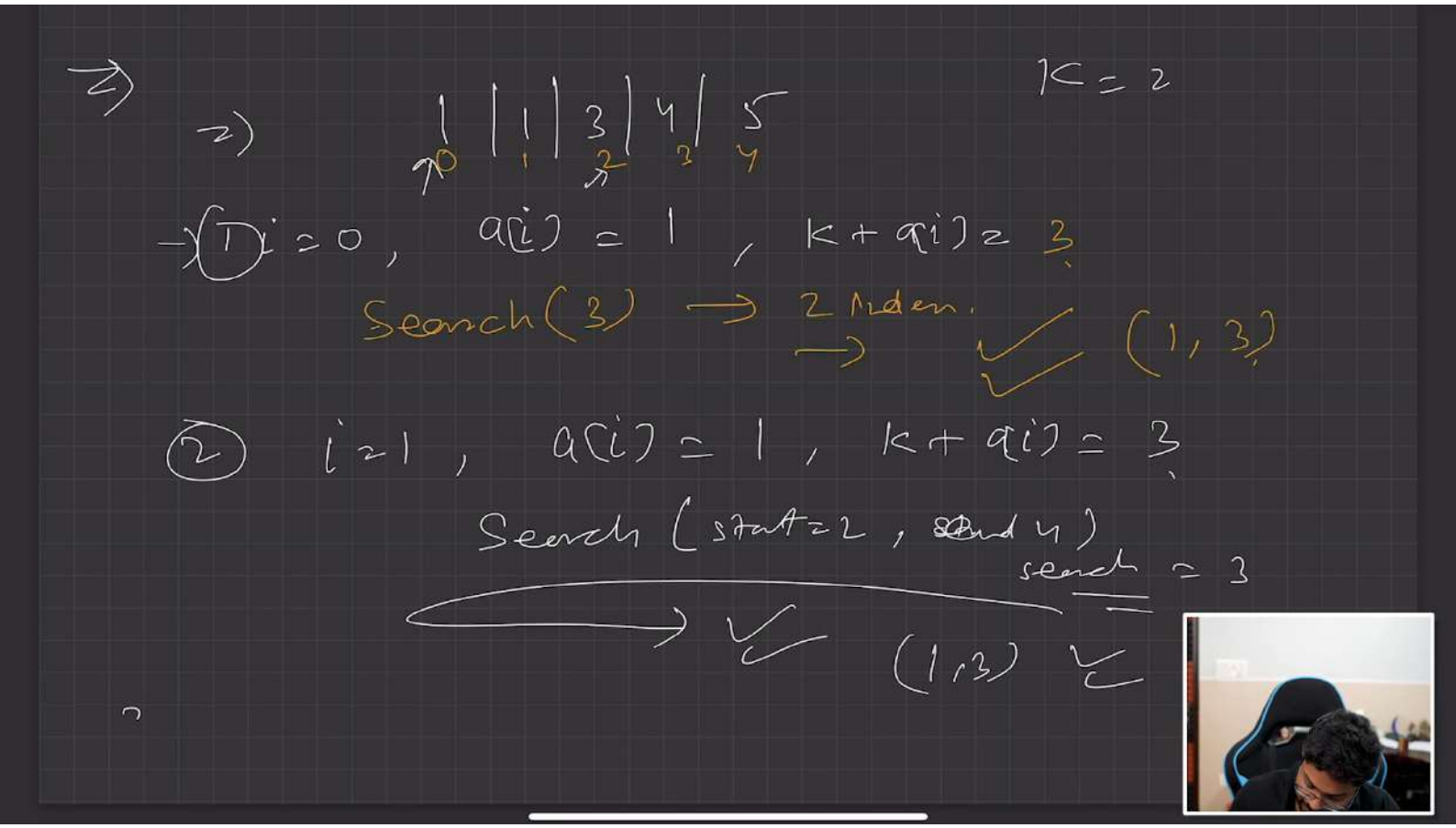
⇒ a[i] , Pair a[j]

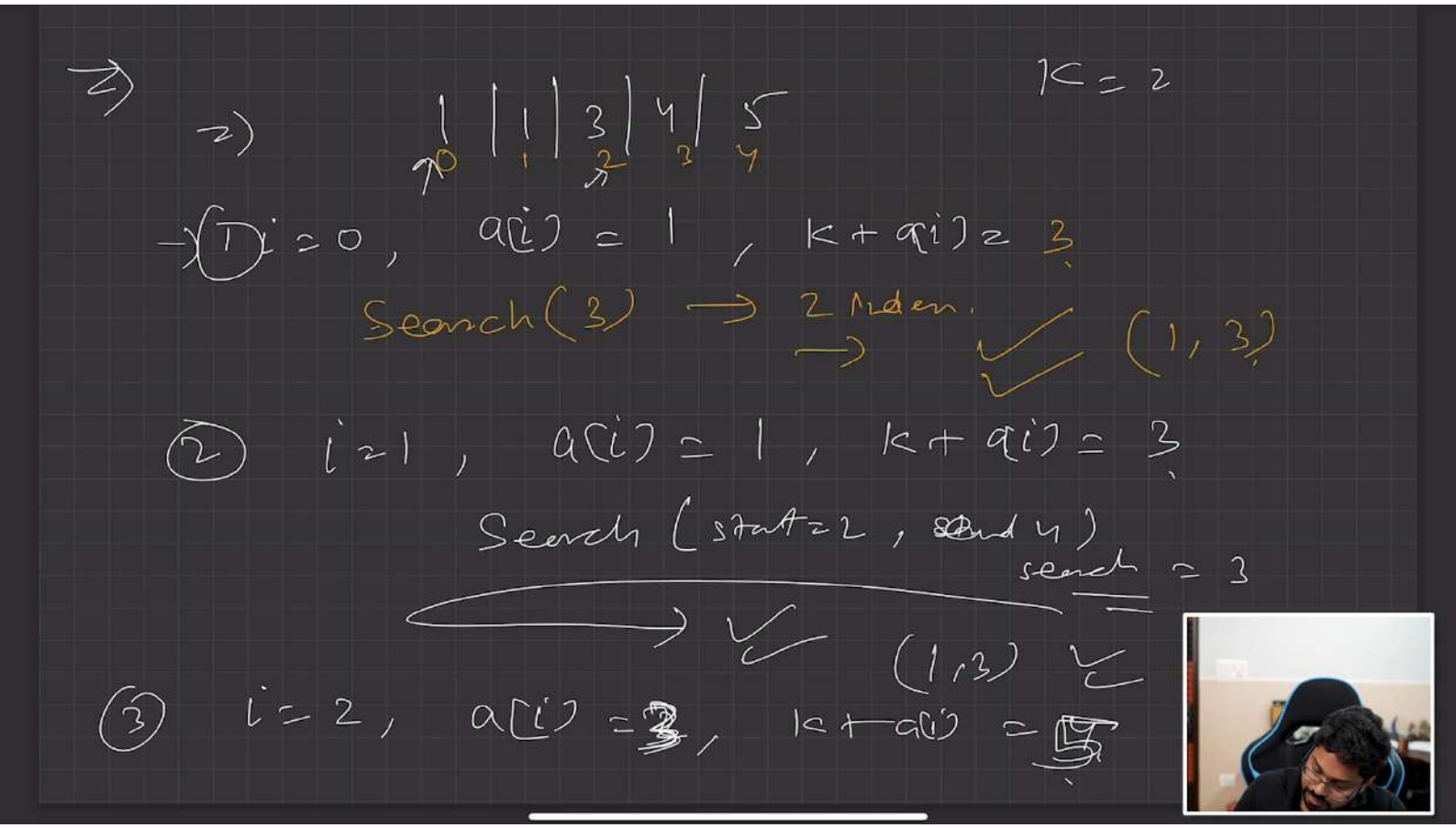
↳ a[j] − a[i] = k

a[j] = k + a[i].

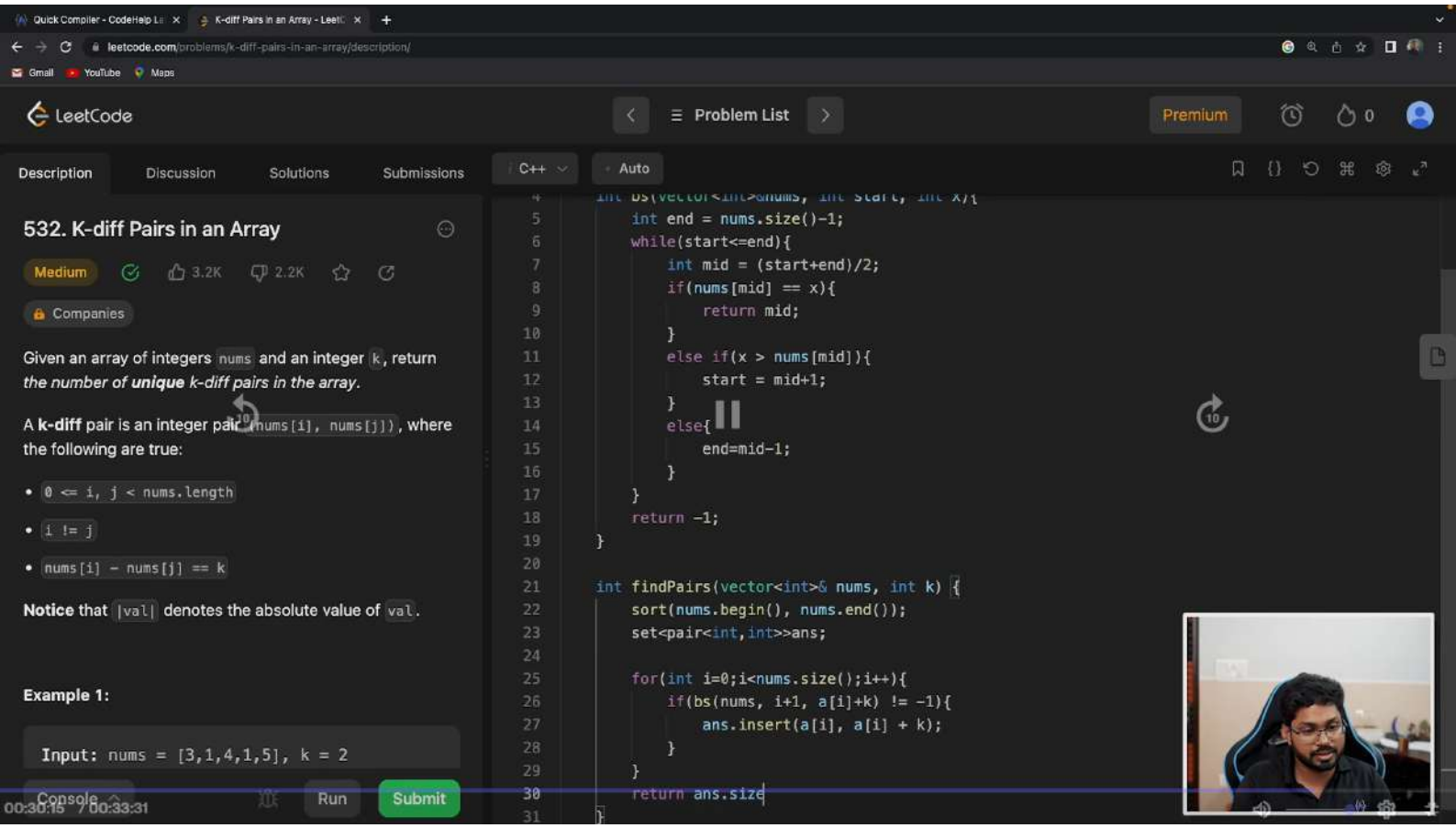
Reframe

a[i] ∈ array a[j] + k ?


k = 2
1 | 1 | 3 | 4 | 5
0 1 2 3 4
① i = 0, a[i] = 1, k + a[i] = 3
Search(3) → 2 Iden. → (1, 3)
② i = 1, a[i] = 1, k + a[i] = 3
Search (start = 2, end 4)
search = 3
(1, 3)


k = 2
1 | 1 | 3 | 4 | 5
0 1 2 3 4
① i = 0, a[i] = 1, k + a[i] = 3
Search(3) → 2 Index.
→ (1, 3)
② i = 1, a[i] = 1, k + a[i] = 3
Search (start = 2, end 4)
search = 3
(1, 3)
③ i = 2, a[i] = 3, k + a[i] = 5

(4) i => , a[i], 4 k + a[i] = 6.


Quick Compiler - CodeHelp La
K-diff Pairs in an Array - LeetC
leetcode.com/problems/k-diff-pairs-in-an-array/description/
Gmail
YouTube
Maps
LeetCode
Problem List
Premium
Description
Discussion
Solutions
Submissions
C++
Auto
532. K-diff Pairs in an Array
Medium
3.2K
2.2K
Companies
Given an array of integers nums and an integer k, return the number of unique k-diff pairs in the array.
A k-diff pair is an integer pair (nums[i], nums[j]), where the following are true:
0 <= i, j < nums.length
i != j
nums[i] - nums[j] == k
Notice that |val| denotes the absolute value of val.
Example 1:
Input: nums = [3,1,4,1,5], k = 2
Console
Run
Submit
00:30:15 / 00:33:31
int bs(vector<int>&nums, int start, int x){
int end = nums.size()-1;
while(start<=end){
int mid = (start+end)/2;
if(nums[mid] == x){
return mid;
}
else if(x > nums[mid]){
start = mid+1;
}
else{
end=mid-1;
}
}
return -1;
}
int findPairs(vector<int>& nums, int k) {
sort(nums.begin(), nums.end());
set<pair<int,int>>ans;
for(int i=0;i<nums.size();i++){
if(bs(nums, i+1, a[i]+k) != -1){
ans.insert(a[i], a[i] + k);
}
}
return ans.size
}

Quick Compiler - CodeHelp L... | K-diff Pairs in an Array - Leet...

leetcode.com/problems/k-diff-pairs-in-an-array/description/

Gmail YouTube Maps

LeetCode | Problem List | Premium | 0

Description | Discussion | Solutions | Submissions

532. K-diff Pairs in an Array

Medium 3.2K 2.2K

Companies

Given an array of integers `nums` and an integer `k`, return *the number of **unique** k-diff pairs in the array*.

A **k-diff** pair is an integer pair `(nums[i], nums[j])`, where the following are true:

- `0 <= i, j < nums.length`
- `i != j`
- `nums[i] - nums[j] == k`

Notice that `|val|` denotes the absolute value of `val`.

Example 1:

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

Console | Run | Submit

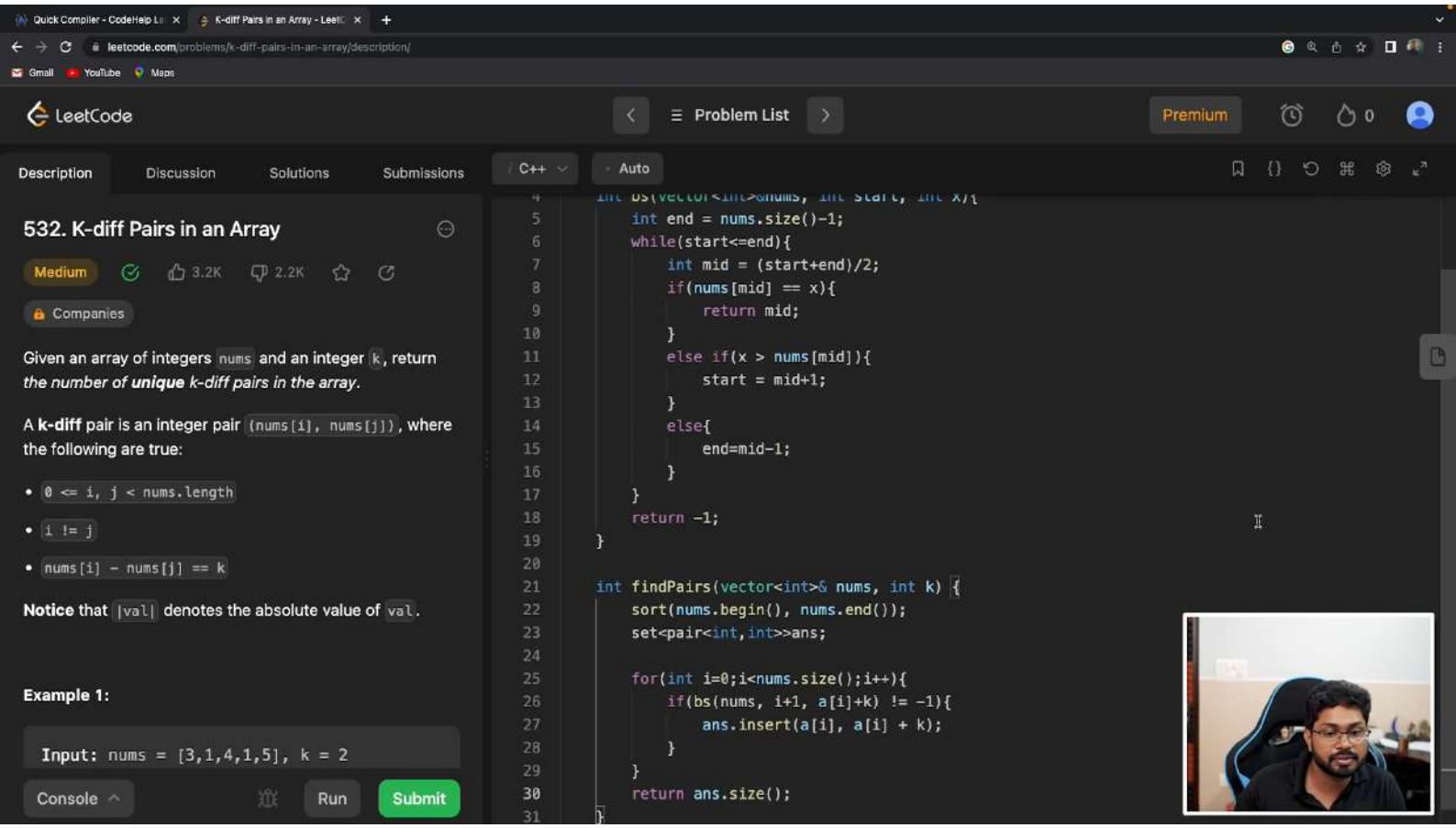
C++ | Auto

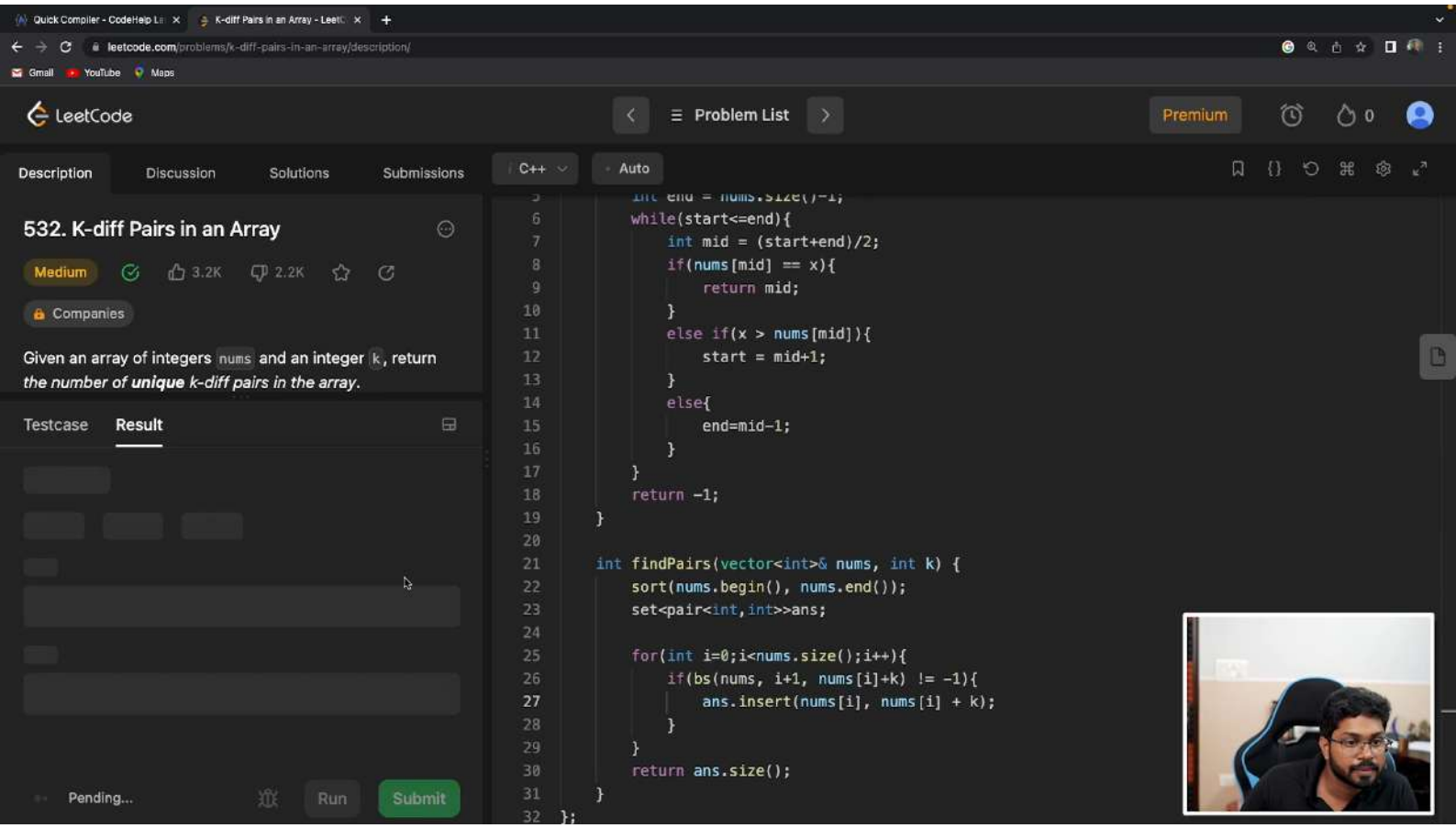
```cpp
public:

    int bs(vector<int>&nums, int start, int x){
        int end = nums.size()-1;
        while(start<=end){
            int mid = (start+end)/2;
            if(nums[mid] == x){
                return mid;
            }
            else if(x > nums[mid]){
                start = mid+1;
            }
            else{
                end=mid-1;
            }
        }
        return -1;
    }

    int findPairs(vector<int>& nums, int k) {
        sort(nums.begin(), nums.end());
        set<pair<int,int>>ans;

        for(int i=0;i<nums.size();i++){
            if(bs(nums, i+1, a[i]+k) != -1){
                ans.insert(a[i])
            }
        }
```


Quick Compiler - CodeHelp L
K-diff Pairs in an Array - Leet
leetcode.com/problems/k-diff-pairs-in-an-array/description/
Gmail
YouTube
Maps
LeetCode
Problem List
Premium
0
Description
Discussion
Solutions
Submissions
C++
Auto
532. K-diff Pairs in an Array
Medium
3.2K
2.2K
Companies
Given an array of integers nums and an integer k, return the number of unique k-diff pairs in the array.
A k-diff pair is an integer pair (nums[i], nums[j]), where the following are true:
0 <= i, j < nums.length
i != j
nums[i] - nums[j] == k
Notice that |val| denotes the absolute value of val.
Example 1:
Input: nums = [3,1,4,1,5], k = 2
Console
Run
Submit
int bs(vector<int>&nums, int start, int x){
int end = nums.size()-1;
while(start<=end){
int mid = (start+end)/2;
if(nums[mid] == x){
return mid;
}
else if(x > nums[mid]){
start = mid+1;
}
else{
end=mid-1;
}
}
return -1;
}
int findPairs(vector<int>& nums, int k) {
sort(nums.begin(), nums.end());
set<pair<int,int>>ans;
for(int i=0;i<nums.size();i++){
if(bs(nums, i+1, a[i]+k) != -1){
ans.insert(a[i], a[i] + k);
}
}
return ans.size();
}


Quick Compiler - CodeHelp La
K-diff Pairs in an Array - Leet
leetcode.com/problems/k-diff-pairs-in-an-array/description/
Gmail
YouTube
Maps
LeetCode
Problem List
Premium
0
Description
Discussion
Solutions
Submissions
C++
Auto
532. K-diff Pairs in an Array
Medium
3.2K
2.2K
Companies
Given an array of integers nums and an integer k, return the number of unique k-diff pairs in the array.
Testcase
Result
Pending...
Run
Submit
int end = nums.size()-1;
while(start<=end){
int mid = (start+end)/2;
if(nums[mid] == x){
return mid;
}
else if(x > nums[mid]){
start = mid+1;
}
else{
end=mid-1;
}
}
return -1;
}
int findPairs(vector<int>& nums, int k) {
sort(nums.begin(), nums.end());
set<pair<int,int>>ans;
for(int i=0;i<nums.size();i++){
if(bs(nums, i+1, nums[i]+k) != -1){
ans.insert(nums[i], nums[i] + k);
}
}
return ans.size();
}
};

(4) i => a(i), 4 K + a(i) = 6.

n(logn)