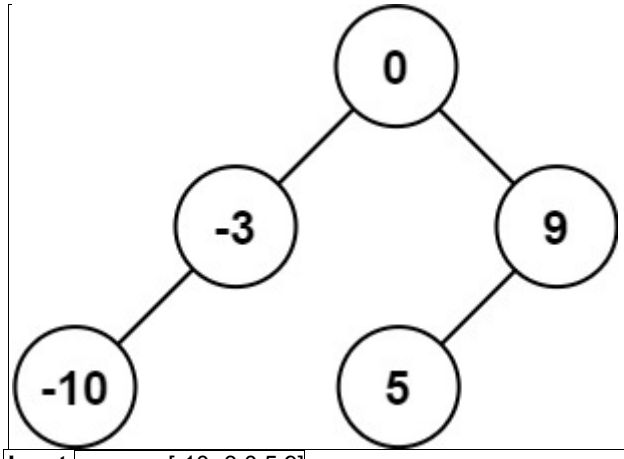
Leetcode Problem 1. (Easy)

Convert Sorted Array to Binary Search Tree

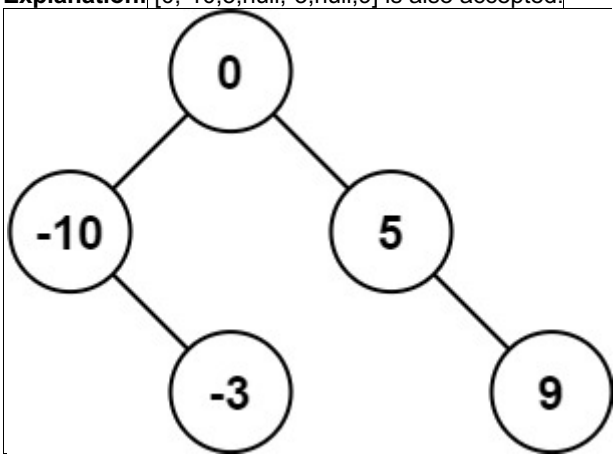
Given an integer array nums where the elements are sorted in **ascending order**, convert *it to a*

height-balanced
binary search tree.

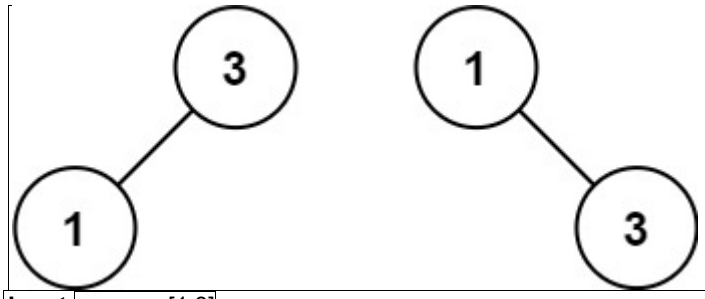
Example 1:



Input: nums = [-10,-3,0,5,9]
Output: [0,-3,9,-10,null,5]
Explanation: [0,-10,5,null,-3,null,9] is also accepted:



Example 2:



Input: nums = [1,3]
Output: [3,1]
Explanation: [1,null,3] and [3,1] are both height-balanced BSTs.

Constraints:

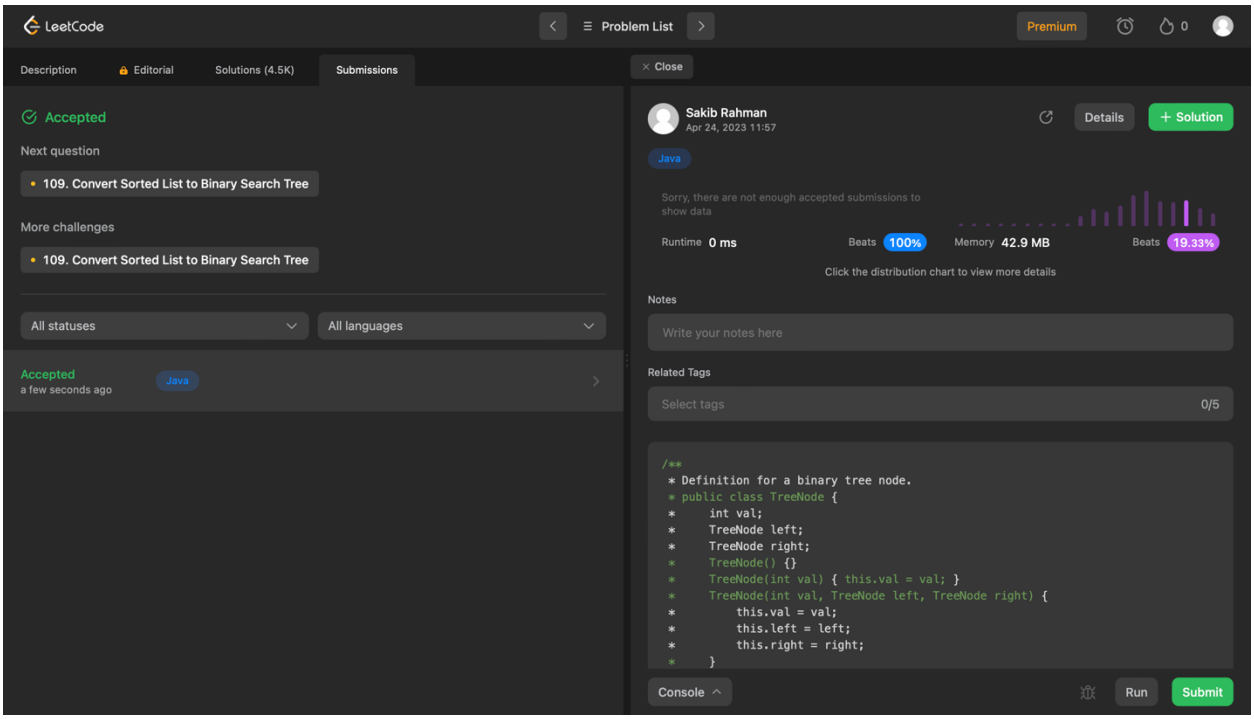
- $1 <= nums.length <= 10^4$
- $-10^4 <= nums[i] <= 10^4$
- nums is sorted in a **strictly increasing** order.

Link: https://leetcode.com/problems/convert-sorted-array-to-binary-search-tree/

```
class Solution {
    public TreeNode sortedArrayToBST(int[] nums) {

  return buildTree(nums, 0, nums.length - 1);
}

private TreeNode buildTree(int[] nums, int left, int right) {
    if (left > right) {
        return null;
    }
    int mid = (left + right) / 2;
    TreeNode root = new TreeNode(nums[mid]);
    root.left = buildTree(nums, left, mid - 1);
    root.right = buildTree(nums, mid + 1, right);
    return root;
}
}
```


LeetCode
Problem List
Premium
0
Description
Editorial
Solutions (4.5K)
Submissions
Close
Accepted
Next question
109. Convert Sorted List to Binary Search Tree
More challenges
109. Convert Sorted List to Binary Search Tree
All statuses
All languages
Accepted
a few seconds ago
Java
Sakib Rahman
Apr 24, 2023 11:57
Details
+ Solution
Java
Sorry, there are not enough accepted submissions to show data
Runtime 0 ms
Beats 100%
Memory 42.9 MB
Beats 19.33%
Click the distribution chart to view more details
Notes
Write your notes here
Related Tags
Select tags
0/5
/**
* Definition for a binary tree node.
* public class TreeNode {
* int val;
* TreeNode left;
* TreeNode right;
* TreeNode() {}
* TreeNode(int val) { this.val = val; }
* TreeNode(int val, TreeNode left, TreeNode right) {
* this.val = val;
* this.left = left;
* this.right = right;
* }
Console
Run
Submit