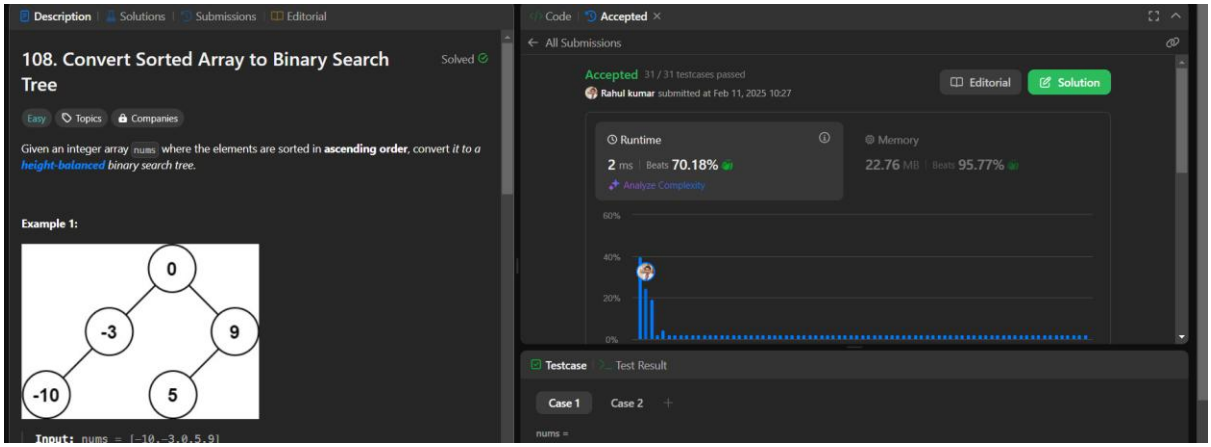
[108. Convert Sorted Array to Binary Search Tree]



Code:

```
/**
 * Definition for a binary tree node.
 * struct TreeNode {
 *     int val;
 *     TreeNode *left;
 *     TreeNode *right;
 *     TreeNode() : val(0), left(nullptr), right(nullptr) {}
 *     TreeNode(int x) : val(x), left(nullptr), right(nullptr) {}
 *     TreeNode(int x, TreeNode *left, TreeNode *right) : val(x), left(left),
 * right(right) {}
 * };
 */
class Solution {
public:
    TreeNode* buildBST(vector<int>& nums, int left, int right) {
        if (left > right)
            return nullptr;

        int mid = left + (right - left) / 2;
        TreeNode* root = new TreeNode(nums[mid]);
```

```
        root->left = buildBST(nums, left, mid - 1);
        root->right = buildBST(nums, mid + 1, right);

        return root;
    }

    TreeNode* sortedArrayToBST(vector<int>& nums) {
        return buildBST(nums, 0, nums.size() - 1);
    }
};
```