Problem List

Run Submit

Description | Editorial | Submissions | Solutions

270. Closest Binary Search Tree Value

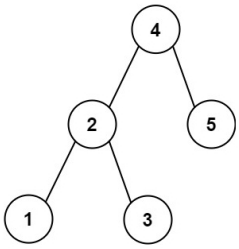
Premium

Solved

Easy | Topics | Companies

Given the `root` of a binary search tree and a `target` value, return *the value in the BST that is closest to the* `target`. If there are multiple answers, print the smallest.

Example 1:




```
Input: root = [4,2,5,1,3], target = 3.714286
Output: 4
```

Example 2:

```
Input: root = [1], target = 4.428571
Output: 1
```

Constraints:

- The number of nodes in the tree is in the range `[1, 10^4]`.
- `0 <= Node.val <= 10^9`
- `-10^9 <= target <= 10^9`

Seen this question in a real interview before? 1/5

Yes No

Accepted 341.6K | Submissions 662K | Acceptance Rate 51.6%

Topics

Companies

Similar Questions

Discussion (12)

1.8K 12

Code

Python3 • Auto

```python
# Definition for a binary tree node.
# class TreeNode:
#     def __init__(self, val=0, left=None, right=None):
#         self.val = val
#         self.left = left
#         self.right = right
class Solution:
    def closestValue(self, root: Optional[TreeNode], target: float) -> int:
        nearest = float("inf")

        while root:
            if root.val == target: return root.val
            if abs(root.val-target) < abs(nearest-target): nearest = root.val
            if abs(root.val-target) == abs(nearest-target): nearest = min(root.val,nearest)

            if root.val > target:
                root = root.left
            else:
                root = root.right

        return nearest
```

Saved Ln 10, Col 1

Testcase | Test Result

Accepted Runtime: 36 ms

• Case 1 • Case 2

Input

root =
[4,2,5,1,3]

target =
3.714286

Output