lee215 ★ 47718 Last Edit: July 13, 2019 10:09 PM 1.7K VIEWS

20

**Complexity**

Time `O(N)`
Space `O(H)`

**Python:**

```
def maximumAverageSubtree(self, root):
    self.res = 0
    def helper(root):
        if not root: return [0, 0.0]
        n1, s1 = helper(root.left)
        n2, s2 = helper(root.right)
        n = n1 + n2 + 1
        s = s1 + s2 + root.val
        self.res = max(self.res, s / n)
        return [n, s]
    helper(root)
    return self.res
```



Type comment here... (Markdown is supported)

Post

tensor-flower ★ 551 July 18, 2019 12:46 PM

intuition
https://www.youtube.com/watch?v=NUIsvZhV4cs&feature=youtu.be


Read More

2 Show 1 reply Reply

tonghuikang ★ 148 July 13, 2019 10:09 PM

make sure it is python3 :/

1 Show 3 replies Reply

SherlockZ ★ 1 Last Edit: July 14, 2019 12:47 PM

Java version with explanation.

the `helper` function return a double array:

```
[0]: sum of subtree's val
[1]: number of nodes in subtree
```

every recursive process pass this result array to current node's root.

```
private double res = 0;
```

Read More

0 Reply