

Complexity Analysis of *calculateMaterialTotals*

1. *calculateMaterialTotals*

- **Initialization:** Creating the materialTotals map: **$O(1)$** .
- **Recursive Traversal (*calculateMaterialTotalsRecursive*):** The method visits every node once, so the time complexity is **$O(N)$** , where **N** is the number of nodes in the tree.
- **Displaying Results:** Iterating over the materialTotals map: **$O(M)$** , where **M** is the number of unique materials.
- **Overall Time Complexity:** **$O(N)$** .

2. *calculateMaterialTotalsRecursive*

- **Processing Each Node:** Updating the materialTotals map is **$O(1)$** .
- **Recursion:** The method processes each node and its children recursively. Worst-case complexity is **$O(N)$** for traversing the entire tree.
- **Overall Time Complexity:** **$O(N)$** .

Summary

Method	Time Complexity (Best Case)	Time Complexity (Worst Case)	Space Complexity
<i>calculateMaterialTotals(TreeNode)</i>	$O(N)$	$O(N)$	$O(N)$
<i>calculateMaterialTotalsRecursive</i>	$O(1)$	$O(N)$	$O(N)$