339. Nested List Weight Sum

April 7, 2016 | 31.4K views

Average Rating: 4.75 (36 votes)

Given a nested list of integers, return the sum of all integers in the list weighted by their depth.

Each element is either an integer, or a list -- whose elements may also be integers or other lists.

Example 1:

```
Input: [[1,1],2,[1,1]]
Output: 10
Explanation: Four 1's at depth 2, one 2 at depth 1.
```

Example 2:

```
Input: [1,[4,[6]]]
Output: 27
Explanation: One 1 at depth 1, one 4 at depth 2, and one 6 at depth 3; 1 + 4*2 + 6*3 =
```

Summary

This is a very simple recursion problem and is a nice introduction to Depth-first Search (DFS).

Solution

Depth-first Traversal [Accepted]

Algorithm

Because the input is nested, it is natural to think about the problem in a recursive way. We go through the list of nested integers one by one, keeping track of the current depth d. If a nested integer is an integer n, we calculate its sum as $n \times d$. If the nested integer is a list, we calculate the sum of this list recursively using the same process but with depth $d + 1$.

Java

```
/**
 * // This is the interface that allows for creating nested lists.
 * // You should not implement it, or speculate about its implementation
 * public interface NestedInteger {
 *
 *     // @return true if this NestedInteger holds a single integer,
 *     // rather than a nested list.
 *     public boolean isInteger();
 *
 *     // @return the single integer that this NestedInteger holds,
 *     // if it holds a single integer
 *     // Return null if this NestedInteger holds a nested list
 *     public Integer getInteger();
 *
 *     // @return the nested list that this NestedInteger holds,
 *     // if it holds a nested list
 *     // Return null if this NestedInteger holds a single integer
 *     public List<NestedInteger> getList();
 * }
 */
public int depthSum(List<NestedInteger> nestedList) {
    return depthSum(nestedList, 1);
}

public int depthSum(List<NestedInteger> list, int depth) {
    int sum = 0;
    for (NestedInteger n : list) {
        if (n.isInteger()) {
            sum += n.getInteger() * depth;
        } else {
            sum += depthSum(n.getList(), depth + 1);
        }
    }
    return sum;
}
```

Complexity Analysis

The algorithm takes $O(N)$ time, where N is the total number of nested elements in the input list. For example, the list `[ [[[[1]]]], 2 ]` contains 4 nested lists and 2 nested integers (1 and 2), so $N = 6$.

In terms of space, at most $O(D)$ recursive calls are placed on the stack, where D is the maximum level of nesting in the input. For example, $D = 2$ for the input `[[1,1],2,[1,1]]`, and $D = 3$ for the input `[1,[4,[6]]]`.

Analysis written by: @noran

Rate this article:

Previous Next

Comments: 10

Sort By

Type comment here... (Markdown is supported)

Preview Post

hero_afei ★ 2 November 16, 2016 8:32 PM

My non recursive solution:
public class Solution {
public int depthSum(List nestedList) {
int ans = 0, depth = 1;
Queue<List> queue = new LinkedList<>();

Read More

2 Share Reply

mian_hashim ★ 0 July 10, 2019 11:06 PM

Can't we use BFS in this problem? Since technically this is a level order traversal?

0 Share Reply

jiangyucara ★ 105 January 30, 2019 8:59 AM

Can anyone explain why do we do the method overload here?

0 Share Reply

SHOW 3 REPLIES

luv_ahuja4 ★ 3 May 13, 2018 3:51 AM

```
/**
 * // This is the interface that allows for creating nested lists.
 * // You should not implement it, or speculate about its implementation
 * public interface NestedInteger {
```

Read More

0 Share Reply

yolo31 ★ 140 October 17, 2017 4:35 AM

The time complexity mentioned here is incorrect. It should be O(N) where N is total number of elements or Integers

0 Share Reply

SHOW 1 REPLY

AmoghR ★ 0 May 2, 2020 1:08 PM

Time complexity should be O(N+L) to make it more precise, where N is the total number of integers and L is the total number of nested lists. L is not really a constant factor and should not be ignored.

0 Share Reply

hackyhack333 ★ 116 March 17, 2020 7:19 AM

Why doesn't isInteger() work in python? says no attribute isInteger...

0 Share Reply

SHOW 1 REPLY

Mrmagician ★ 178 March 13, 2020 3:34 PM

My iterative solution:

```
stack = [[1, nestedList]]
        count = 0
while len(stack):
```

Read More

0 Share Reply

zabrosov ★ 0 January 9, 2020 3:28 PM

```
class Solution(object):
    def depthSum(self, nestedList):
        """
        :type nestedList: [NestedInteger]
```

Read More

0 Share Reply

XiangyuL926 ★ 127 October 29, 2019 1:34 PM

iterative... even slow
cpp:

```
class Solution {
public:
```

Read More

0 Share Reply