

1807 - DS_2019Fall_HW_3

Scoreboard (/contest/scoreboard/1807/)

Time		
2019/10/28 10:10:00	8days, 08:06:58	2019/11/11 23:59:00

Clarification						
#	Problem	Asker	Description	Reply	Replier	Reply Time

Overview

Problem ▾

12449 - DS_2019Fall_HW_3

Status (/status/?pid=12449) | Limits

Submit (/users/submit/12449)

Description

In this homework, you are asked to implement the following functions.

1. Construct a binary tree
2. Tree traversal
3. Evaluate the height of a binary tree
4. Determine whether a binary tree is complete or not
5. Determine whether a binary tree is foldable or not
6. QQ addition
7. Delete all the leaves in a binary tree

You are not allow to use STL.

Implement your own stack to parse s-expression.

Input

Each test case contains multiple inputs. Each input is a string of s-expression.

Please note:

- 1) The length of each s-expression is at most 10000000.
- 2) The number of nodes in each tree is at most 1200000.
- 3) Each node's weight is between -100000 and 100000.

Output

For each input s-expression, ouput its following things separated by a newline symbol.

- Pre-order traversal[#]
- In-order traversal[#]
- Post-order traversal[#]
- The height of the binary tree
- Whether the binary tree is complete or not
- Whether the binary tree is complete or not
- QQ addition
- Pre-order traversal after deleting all the leaves[#]
- In-order traversal after deleting all the leaves[#]
- Post-order traversal after deleting all the leaves[#]

[#]Note: if the tree is empty, print nothing but a new line character for the pre/in/post –order traversal results

Sample Input

Download (data:text/plain;charset=utf-8,(1(2(4())(5()))(3(6())(7()))))

```
()
(1(2(4())(5()))(3(6())(7()))))
```

Sample Output

Download (data:text/plain;charset=utf-8,%0D%0A%0D%0A%0D%0A0%0D%0AComplete%0D%0AFoldable%0D%0A0%0D%0A%0D%0A%0D%0A%0D%0A1%202%

```
0
Complete
Foldable
0

1 2 4 5 3 6 7
4 2 5 1 6 3 7
4 5 2 6 7 3 1
3
Complete
Foldable
23
1 2 3
2 1 3
2 3 1
```

Discuss