# 描述

完全二叉树是每一层（除最后一层外）都是完全填充（即，结点数达到最大）的，并且所有的结点都尽可能地集中在左侧。

设计一个用完全二叉树初始化的数据结构 CBTInserter，它支持以下几种操作：

CBTInserter(TreeNode root) 使用头结点为 root 的给定树初始化该数据结构；

CBTInserter.insert(int v) 将 TreeNode 插入到存在值为 node.val = v  的树中以使其保持完全二叉树的状态，并返回插入的 TreeNode 的父结点的值；

CBTInserter.get\_root() 将返回树的头结点。

**示例 1：**

输入：inputs = ["CBTInserter","insert","get\_root"], inputs = [[[1]],[2],[]]

输出：[null,1,[1,2]]

**示例 2：**

输入：inputs = ["CBTInserter","insert","insert","get\_root"], inputs = [[[1,2,3,4,5,6]],[7],[8],[]]

输出：[null,3,4,[1,2,3,4,5,6,7,8]]

**提示：**

最初给定的树是完全二叉树，且包含 1 到 1000 个结点。

每个测试用例最多调用 CBTInserter.insert  操作 10000 次。

给定结点或插入结点的每个值都在 0 到 5000 之间。

# 分析

# 代码