

二叉搜索树插入

(insert_bst.cpp/c/pas)

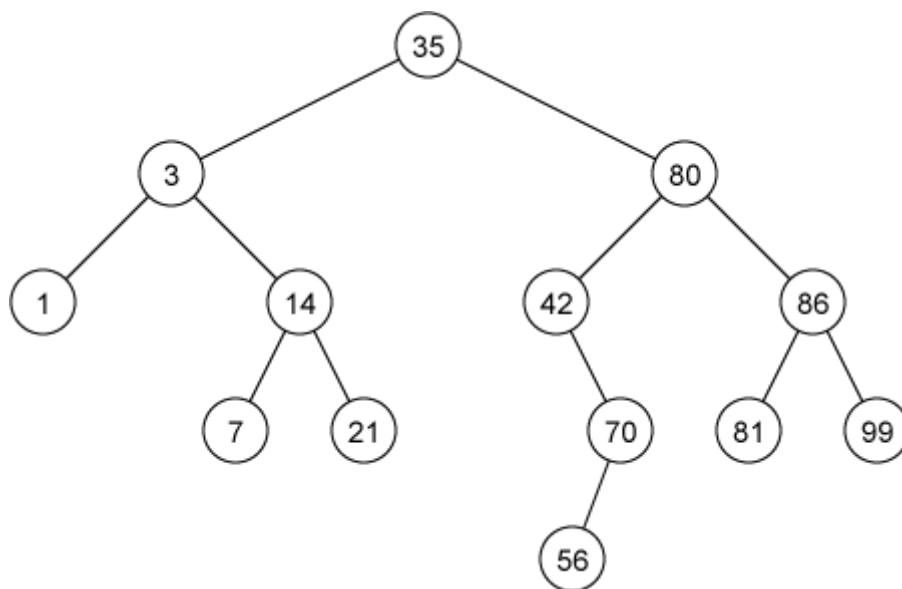
Time Limit : 2 sec , Memory Limit : 128MB

Description

搜索树是一种可以进行插入、搜索、删除等操作的数据结构，可以用作字典或优先级队列。二叉搜索树属于最基本的搜索树。二叉搜索树的各结点均拥有键值，且满足下述性质：

- 设 x 为二叉搜索树的结点。如果 y 是 x 左子树中的结点，那么 y 的键值 $y.key \leq x.key$ 。如果 y 是 x 右子树中的结点，那么 x 的键值 $x.key \leq y.key$ 。

下图是二叉搜索树的例子。



下述伪代码所示的 `insert` 用于在二叉搜索树 T 中插入新值 v 。 `insert` 将键值为 v ，左子树为 `NIL`，右子树为 `NIL` 的点 z 作为实参，插入 T 的恰当位置。

```
1 insert(T, z)

2     y = NIL // parent of x

3     x = 'the root of T'

4     while x ≠ NIL

5         y = x // set the parent

6         if z.key < x.key

7             x = x.left // move to the left child

8         else
```

```
9         x = x.right // move to the right child

10     z.p = y

11

12     if y == NIL // T is empty

13         'the root of T' = z

14     else if z.key < y.key

15         y.left = z // z is the left child of y

16     else

17         y.right = z // z is the right child of y
```

请按照上述伪代码实现插入算法。

Input

第一行输入命令数 m 。

接下来 m 行以 **insert k** 或者 **print** 的格式输入命令，每个命令占 **1** 行。

Output

每执行 **1** 次 **print** 命令后,就分别输出中序遍历算法和前序遍历算法所得到的键值序列,每个序列占 **1** 行。
每个键值前输出 **1** 个空格。

Constraints

- 命令数 $\leq 500,000$
 - **print** 命令数 ≤ 10
 - $-2,000,000,000 \leq k \leq 2,000,000,000$
 - 树的高度不会超过 **100**
 - 二叉搜索树各结点的键值不重复
-

Sample Input 1

```
8
insert 30
insert 88
insert 12
insert 1
insert 20
insert 17
insert 25
print
```

Sample Output 1

```
1 12 17 20 25 30 88

30 12 1 20 17 25 88
```