Given the root of a binary tree, then value v and depth d, you need to add a row of nodes with value v at the given depth d. The root node is at depth 1.

The adding rule is: given a positive integer depth d, for each NOT null tree nodes N in depth d-1, create two tree nodes with value v as N's left subtree root and right subtree root. And N's **original left subtree** should be the left subtree of the new left subtree root, its **original right subtree** should be the right subtree of the new right subtree root. If depth d is 1 that means there is no depth d-1 at all, then create a tree node with value **v** as the new root of the whole original tree, and the original tree is the new root's left subtree.

**Example 1:**

**Input:**

A binary tree as following:

4

/ \

2 6

/ \ /

3 1 5

**v = 1**

**d = 2**

**Output:**

4

/ \

1 1

/ \

2 6

/ \ /

3 1 5

**Example 2:**

**Input:**

A binary tree as following:

4

/

2

/ \

3 1

**v = 1**

**d = 3**

**Output:**

4

/

2

/ \

1 1

/ \

3 1

**Note:**

1. The given d is in range [1, maximum depth of the given tree + 1].
2. The given binary tree has at least one tree node.