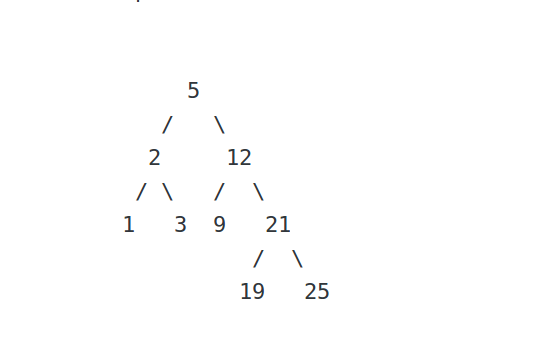
Given the **root of a binary search tree** and a number **n**, find the greatest number in the binary search tree that is less than or equal to **n**.



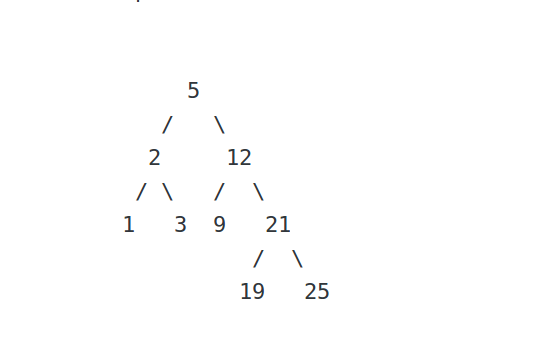
n = 24

**Output :**   
21

**Explanation :** The greatest element in the tree which

is less than or equal to 24, is 21.

(Searching will be like 5->12->21)



n = 4

**Output :**   
3

**Explanation :** The greatest element in the tree which

is less than or equal to 4, is 3.

(Searching will be like 5->2->3)

You don't need to read input or print anything. Your task is to complete the function **findMaxForN()** which takes the **root** of the BST, and the integer **n** as input paramters and returns the **greatest element less than or equal** to **n** in the given BST.

**Expected Time Complexity:**O(Height of the BST)  
**Expected Auxiliary Space:**O(Height of the BST).