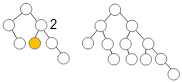
The height of a height-balanced binary tree

A binary tree is *height-balanced* if, for every node, the heights of its two children differ by at most one.

******The first tree on the right above is not balanced because the root’s left subtree has height -1 and its right subtree has height 1. The second tree is balanced.

# Below, we will prove that the height of a balanced tree of n nodes is O(lg n). This is in spite of how “unbalanced” a height-balanced tree can be. Take a look at the second tree to the right. It is height-balanced. It was formed by adding a right child to the rightmost node of the tree to its left and then moving up the tree, adding a node here or there, always to the right, to ensure that the heights of the children of each node differ by at most one. Note that the heights of the leftmost leaf and rightmost leaves differ by 2. If we performed the procedure twice more, they would differ by 3.

Yet, we can still prove that the height of a balanced tree of n nodes is O(n).

# To shorten the verbiage, we use simply *tree* for *binary tree* and *balanced* for *height-balanced.* Further, we use lg n for the base-2 logarithm of n and define:

# min(h) is the minimum number of nodes in a balanced tree of height h.

# We prove the theorem based on two lemmas, which are proved below.

# Theorem: The height of a balanced tree of n nodes is O(lg n).

**Proof**. Lemma 2 below proves that

h < 2 (lg min(h)) + 2

Since min(h) is the minimum number of nodes in a balanced tree of height h, we have

h < 2 (lg n) + 2 for *any* balanced tree of height h with n > 1 nodes.

Therefore, h is O(lg n).

Q.E.D. (Quit.End.Done.)

That was the easy part of the proof! We now turn to the hard part, proving a lower bound on the number of nodes in a tree based on its height:

**Lemma 1**. A balanced tree with height h > 0 has at least 2h/2 - 1 nodes.

**Proof**. The proof is by induction on h.

Case h = 1. A balanced tree of height 1 has at least 2 nodes. We have: 21/2 - 1 < 2, so the theorem holds for h = 1.

Case h = 2. A balanced tree of height 2 has at least 4 nodes —see the first tree at the top of this page.   
We have: 22/2 - 1 = 20 = 1 < 4, and the theorem holds for h = 2.

Case h ≥ 3. Assume the theorem holds for values in the range 1..h-1 and prove it holds for h. Consider a balanced tree of height h with a minimum number of nodes. It has a root and two subtrees. Since the tree has a minimum number of nodes, so do it subtrees.

One of the subtrees has height h-1. Since the subtrees have a minimum number of nodes, the other subtree has height h-2 (see the tree to the right). It can’t have a smaller height because the tree is balanced —for any node, the heights of its subtrees differ by at most 1. From this, we see that:

min(h) = 1 + min(h-1) + min(h-2)

We calculate:

min(h)

= < the above formula>

1 + min(h-1) + min(h-2)

> <arithmetic>

min(h-1) + min(h-2)

> <arithmetic, since min(h-1) > min(h-2)>

2 min(h-2)

> <inductive hypothesis>

2 \* 2(h-2)/2 - 1

> <arithmetic>

2 \* 2h/2 - 2

> <arithmetic>

2h/2 - 1

Q.E.D.

Lemma 2 uses Lemma 1 to give the lower bound on the number of nodes in a balanced tree given its height.

**Lemma 2.** h < 2 lg min(h) + 2

**Proof.** From lemma 1, we have:

min(h) > 2h/2 – 1

= <take lg of both sides>

lg min(h) > h/2 – 1

= <arithmetic>

h/2 < lg min(h) + 1

= <arithmetic>

h < 2 lg min(h) + 2