

Number of Nodes in a Perfect Binary Tree

0xHaru

April 20, 2026

1 Definitions

Definition 1 (Height of a binary tree). The **height** of a binary tree is the number of edges on the longest path from the root to a leaf. A tree consisting of a single node has height 0.

Definition 2 (Perfect binary tree). A binary tree is **perfect** if every internal (non-leaf) node has exactly two children and all leaf nodes reside at the same depth.

Figure 1 shows an example of a perfect binary tree of height $h = 3$.

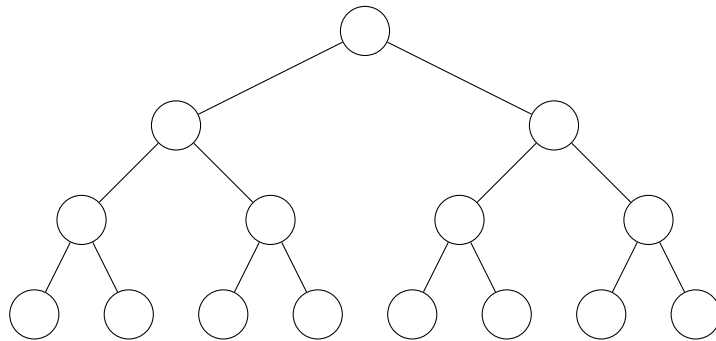


Figure 1: A perfect binary tree of height $h = 3$. Every internal node has exactly two children and all leaves sit at depth 3.

2 Main Theorem

Theorem 1. *A perfect binary tree of height $h \geq 0$ has exactly $2^{h+1} - 1$ nodes.*

The two sections that follow each give a complete proof of Theorem 1.

3 Proof 1 (Induction on the height)

Proof. By induction on h . Let $N(h)$ denote the number of nodes in a perfect binary tree of height h .

Base case ($h = 0$). A perfect tree of height 0 is a single root node with no children, so

$$N(0) = 1 = 2^{0+1} - 1.$$

Inductive hypothesis. Assume, for some fixed $h \geq 1$, that

$$N(h - 1) = 2^{(h-1)+1} - 1 = 2^h - 1.$$

Inductive step. We show $N(h) = 2^{h+1} - 1$.

A perfect binary tree of height h consists of a root together with its left and right subtrees. Because the original tree is perfect, each subtree is itself a perfect binary tree of height $h - 1$ (see Figure 2). Hence

$$\begin{aligned} N(h) &= \underbrace{1}_{\text{root}} + \underbrace{N(h - 1)}_{\text{left subtree}} + \underbrace{N(h - 1)}_{\text{right subtree}} \\ &= 1 + 2N(h - 1) \\ &= 1 + 2(2^h - 1) && \text{(inductive hypothesis)} \\ &= 1 + 2^{h+1} - 2 \\ &= 2^{h+1} - 1. \quad \square \end{aligned}$$

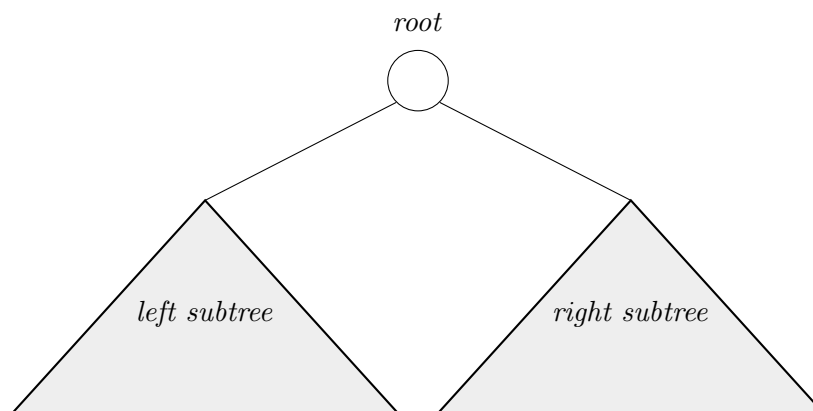


Figure 2: A perfect binary tree of height h decomposed into its root and two perfect subtrees of height $h - 1$, each containing $N(h - 1)$ nodes.

4 Proof 2 (Counting nodes level by level)

Proof. The key observation — visible directly from Figure 3 — is that each level k contains exactly 2^k nodes, so the total is the geometric sum $\sum_{k=0}^h 2^k$. Part 1 establishes that level count; Part 2 evaluates the sum.

Part 1. Let ℓ_k denote the number of nodes at level k . We show $\ell_k = 2^k$ by induction on k .

Base case ($k = 0$). Level 0 contains exactly the root, so $\ell_0 = 1 = 2^0$.

Inductive hypothesis. Assume $\ell_k = 2^k$, for some $0 \leq k < h$.

Inductive step. We show $\ell_{k+1} = 2^{k+1}$.

Since the tree is perfect, every non-leaf node has *exactly* two children. Since $k < h$, every node at level k is internal, so each of the ℓ_k nodes there has exactly two children, all sitting at level $k + 1$. Therefore

$$\ell_{k+1} = 2\ell_k = 2 \cdot 2^k = 2^{k+1}.$$

Figure 3 illustrates the node counts at each level for a perfect tree of height $h = 3$.

Part 2. *Summing over all levels.*

With Part 1 established, summing over all levels $k = 0, 1, \dots, h$ gives:

$$\begin{aligned} \text{Total nodes} &= \sum_{k=0}^h 2^k \\ &= \frac{2^{h+1} - 1}{2 - 1} \quad \left(\text{geometric series: } \sum_{k=0}^n r^k = \frac{r^{n+1} - 1}{r - 1} \right) \\ &= 2^{h+1} - 1. \quad \square \end{aligned}$$

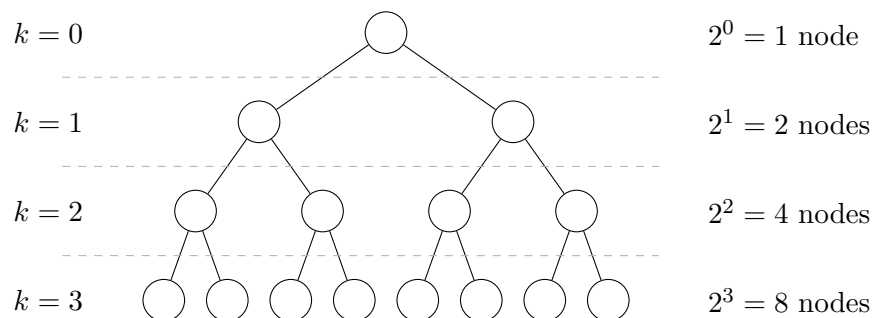


Figure 3: Node counts at each level in a perfect binary tree of height $h = 3$. Each level k contains exactly 2^k nodes, and each node at level k has two children at level $k + 1$.