An example of a proof by structural induction

Define the set $\mathcal{B}$ of *binary trees* as follows:

1. A tree with a single node r is in $\mathcal{B}$; and
2. If r is a node and T_1 and T_2 are binary trees, i.e., $T_1 \in \mathcal{B}$ and $T_2 \in \mathcal{B}$, then the tree $T = (r, T_1, T_2)$ is a binary tree, i.e., T is in $\mathcal{B}$. You should view T as a tree with root r with r having as left child the tree T_1 and as right child the tree T_2.

We now define two functions $|.| : \mathcal{B} \rightarrow \mathbf{N}$ and $h : \mathcal{B} \rightarrow \mathbf{N}$ which respectively return the number of nodes in a tree and the height of the tree.

1. Definition of $|.| : \mathcal{B} \rightarrow \mathbf{N}$:
   (a) $|r| = 1$
   (b) $|(r, T_1, T_2)| = 1 + |T_1| + |T_2|$
2. Definition of $h : \mathcal{B} \rightarrow \mathbf{N}$:
   (a) $h(r) = 0$
   (b) $h((r, T_1, T_2)) = 1 + \max(h(T_1), h(T_2))$.

We will now proof that $\forall T \in \mathcal{B}$: $|T| \leq 2^{h(T)+1} - 1$.

Proof. The proof will be by structural induction.

1. Let T be the tree consisting of a single node r. This is the base case. By definition, $|r| = 1$. By definition $h(r) = 0$. So clearly, $|r| \leq 2^{h(r)+1} - 1$.
2. Now let $T = (r, T_1, T_2)$. Since T_1 and T_2 are "simpler" trees than T, we can assume by *structural induction* that

$$|T_1| \leq 2^{h(T_1)+1} - 1 \qquad \text{(a)}$$

and

$$|T_2| \leq 2^{h(T_2)+1} - 1 \qquad \text{(b).}$$

Therefore,

$$\begin{aligned}
|T| &= |(r, T_1, T_2)| \\
&= 1 + |T_1| + |T_2| \qquad \text{(by definition of } |.|) \\
&\leq 1 + (2^{h(T_1)+1} - 1) + (2^{h(T_2)+1} - 1) \qquad \text{(by (a) and (b) above)} \\
&\leq 2.2^{\max(h(T_1), h(T_2))+1} - 1 \qquad \text{(by simple algebra of max function)} \\
&= 2.2^{h(T)} - 1 \qquad \text{(by definition of the } h \text{ function)} \\
&= 2^{h(T)+1} - 1.
\end{aligned}$$