Trees Tutorial Solutions

13.3b Non-grammar tree induction

Let $T$ be a parity tree; we will prove $T$ has the parity property by induction on its height $h$.

Base: For height 0, $T$ is just a solitary root. That root is also a leaf so it is orange by rule 1 of parity trees. Thus there is an odd number of leaves (1) and the root is orange, so $T$ has the parity property.

(Commentary: You might think you need two base cases here: height 0 for an orange-root case and height 1 for blue-root. However, while including an extra base case doesn’t invalidate the proof, it’s not actually necessary here - to see that, try following through the logic of the induction step below using the concrete height 1 tree plugged in for $T$ everywhere.)

Induction: Suppose that all trees with height less than $h$ have the parity property. Then for tree $T$ with height $h$, consider its left and right subtrees $T_L$ and $T_R$, and let $n_L$ and $n_R$ be the number of leaves in the respective subtrees. Since $T_L$ and $T_R$ have height smaller than $h$, by the IH we know they both have the parity property. (You can not say that they have height $h-1$ - one of them definitely does, but the other could be arbitrarily shorter. This is why it is important that we are using a strong IH.) Now we get four cases:

Case 1: $n_L$ and $n_R$ are both even. Then by the parity property, $T_L$ and $T_R$ both have blue roots. Then by rule 2 of parity trees, $T$ also has a blue root. And we know the total number of leaves is $n_L + n_R$ which is even (because its the sum of two evens), so $T$ has the parity property.

Case 2: $n_L$ and $n_R$ are both odd. Then by the parity property, $T_L$ and $T_R$ both have orange roots. Then by rule 2 of parity trees, $T$ has a blue root. And we know the total number of leaves is $n_L + n_R$ which is even (because its the sum of two odds), so $T$ has the parity property.

Case 3: $n_L$ is even and $n_R$ is odd. Then by the parity property, $T_L$ has a blue root and $T_R$ has an orange root. Then by rule 2 of parity trees, $T$ has an orange root. And we know the total number of leaves is $n_L + n_R$ which is odd (because its the sum of an even and an odd), so $T$ has the parity property.

Case 4: $n_L$ is odd and $n_R$ is even. See case 3 with the roles of $T_L$ and $T_R$ reversed.

Thus $T$ has the parity property in every case, QED.

13.2a Grammar Trees

Proof by induction on the tree height.

Base: Notice that trees from this grammar always have height at least 1. The only ways to produce a tree of height 1 are the third and fourth rules; in each case the tree ends up with one node labeled $a$ and at most one labeled $b$.

Induction: Assume that any tree of height less than some $k > 1$ has at least as many $a$ nodes as $b$s. Now consider a generated tree with height $k$. The root must be labelled $S$ and the grammar rules that can produce trees of height greater than 1 give us two cases for what the children are:

Case 1: The root’s children are labeled $a$, $S$, $b$, and $S$. Let $T_1$ and $T_2$ be the subtrees rooted at the nodes labeled $S$, and let $a_1, a_2, b_1, b_2$ be how many $a$ nodes and $b$ nodes are in
each subtree. Since $T_1$ and $T_2$ have height less than $k$, the IH applies to them, so $a_1 \geq b_1$ and $a_2 \geq b_2$. Putting these two inequalities together and adding one, we establish that $a_1 + a_2 + 1 \geq b_1 + b_2 + 1$. And $a_1 + a_2 + 1$ is just the total number of $a$ nodes in the tree while $b_1 + b_2 + 1$ is the total number of $b$ nodes, so we have shown that there are at least as many $a$s overall as $b$s.

Case 2: The root’s children are labeled $S$, $a$, $S$. The logic here is exactly like case 1 except with one fewer $b$ node, so there are definitely at least as many $a$s as $b$s.

Thus in every case there are at least as many $a$s as $b$s, induction complete.

13.4 Challenge Example

a) Proof by induction on the order $k$ of the tree.

Base: A binomial tree of order 0 is defined to have just 1 = $2^0$ node.

Induction: Let $k \geq 0$ and suppose that for every tree of order $i < k$, a binomial tree of order $i$ has $2^i$ nodes. A binomial tree of order $k$ is built from 2 trees of order $k - 1$, which by the IH have $2^{k-1}$ nodes each. Thus the whole tree has $2^{k-1} + 2^{k-1} = 2^k$ nodes, QED.

b) Proof by induction on the order $k$ of the tree.

Base: A binomial tree of order 0 is defined to have just 1 node at level 0. $\binom{0}{0} = 1$.

Induction: Fix $k \geq 0$ and suppose that for every binomial tree with order $r \leq k$, at each level $i$ the tree has $\binom{r}{i}$ nodes. Now consider a binomial tree of order $k + 1$. By the definition of a binomial tree, it consists of two trees $T_1$ and $T_2$ each of order $k$, where each node in $T_2$ has been ‘shifted down’ one level since $T_2$’s root was connected as the rightmost child of the root of $T_1$. Note that the IH applies to both $T_1$ and $T_2$. Now fix a level $i$. There are three cases:

Case 1: $i = 0$. In this case $T_1$ contributes 1 node to the level and $T_2$ contributes 0, so in total there is $1 = \binom{k+1}{0}$. (Commentary: note that for this case we don’t actually need the IH. This is fine. But if you ever find that your entire inductive step doesn’t use the IH, that would be a major red flag that you’re almost certainly doing something wrong.)

Case 2: $i = k + 1$. In this case $T_1$ has no nodes at level $i$. By the IH, $T_2$ has $\binom{k}{k} = 1$, so in total there is $1 = \binom{k+1}{k+1}$.

Case 3: $0 < i < k + 1$. In this case, by the IH we get $\binom{k}{i}$ nodes from $T_1$ and $\binom{k}{i-1}$ nodes from $T_2$. Thus the total number of nodes at level $i$ is $\binom{k}{i} + \binom{k}{i-1}$. We simplify that as follows: $\binom{k}{i} + \binom{k}{i-1} = \frac{k!}{i!(k-i)!} + \frac{k!}{(i-1)!(k-1-i)!} = \frac{k!(k-i+1) + k!(i)}{i!(k-i)!} = \frac{k!(k+1)}{i!(k-i+1)!} = \frac{(k+1)!}{i!(k-i+1)!} = \binom{k+1}{i}$. So there are $\binom{k+1}{i}$ nodes at level $i$, QED.