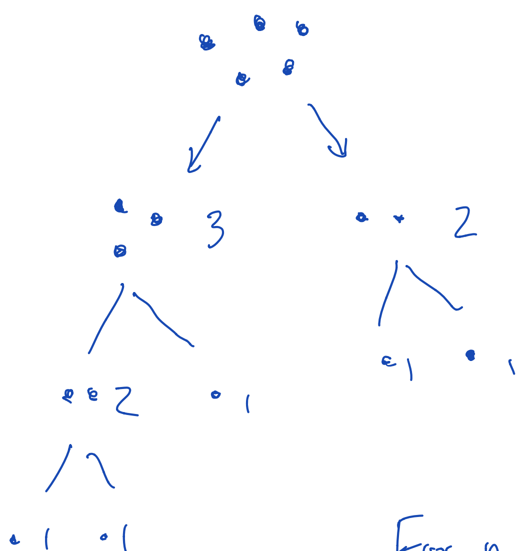
David, Brayden, Chris

Math 251, Fri 22-Oct-2021 -- Fri 22-Oct-2021
Discrete Mathematics
Fall 2021

Friday, October 22nd 2021

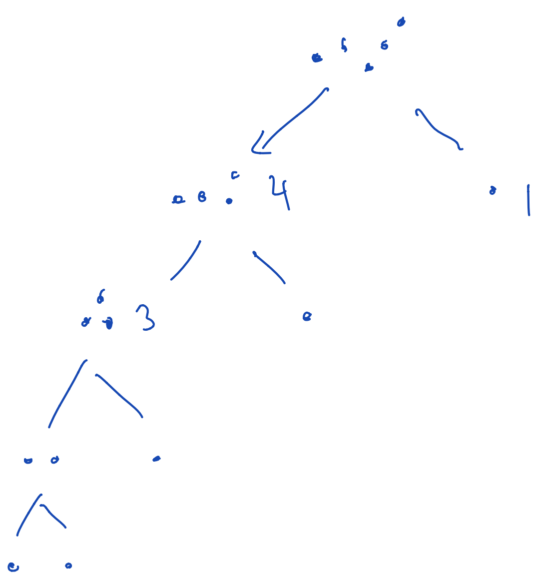
Wk 8, Fr
Topic:: Structural induction
Read:: Rosen 5.3
~~HW[[PS11 due~~ Wed.

$n = 5$



$3 \cdot 2 = 6$
$2 \cdot 1 = 2$
$1 \cdot 1 = 1$
$1 \cdot 1 = 1$
$\overline{\quad 10 \quad}$

For n stones, sum of products is $\dfrac{n(n-1)}{2}$.



$4 \cdot 1 = 4$
$3 \cdot 1 = 3$
$2 \cdot 1 = 2$
$1 \cdot 1 = 1$
$\overline{\quad 10 \quad}$

5.3 recursive definitions and structural induction

Examples of recursive definitions

7. Take Σ to be an alphabet, and Σ^* to be the strings over the alphabet Σ, with empty string λ. Define the **length** function for inputs from Σ^* recursively:

   basis step: $\ell(\lambda) = 0$ (length of empty string)

   recursive step: For $w \in \Sigma^*$ and $x \in \Sigma$, $\ell(wx) = \ell(w) + 1$. (Here wx is the concatenation of w followed by x.)

   Claim: For $w, z \in \Sigma^*$, $\ell(wz) = \ell(w) + \ell(z)$.

   Prove using *structural induction*, a technique for demonstrating properties that hold for elements of a recursively-defined set.

> **Definition 1:** In **structural induction**, one shows
>
> **basis step**: the result holds for all elements specified in the basis step of the recursive definition for the set.
>
> induction ~~**recursive**~~ **step**: if the statement is true for each element used to construct new elements in the recursive step of the definition, then the result holds for these new elements.

Prove statements that apply to objects defined recursively.

9. We define recursively various collections of **rooted trees**. Let the set $\mathcal{R}$ be defined as follows:

   basis step: A single vertex $r \in \mathcal{R}$.

   recursive step: Suppose $T_1, T_2, \ldots T_n \in \mathcal{R}$ are disjoint having roots $r_1, \ldots, r_n$, respectively. The graph formed by taking as root a vertex r not in any of $T_1, \ldots, T_n$, and adding an edge from r to each of the vertices $r_1, \ldots, r_n$ is also in $\mathcal{R}$. Call the resulting tree $T_1 \cdot T_2 \cdots T_n$.

   Let the set $\mathcal{E}$ be defined as follows:

   basis step: The empty set is in $\mathcal{E}$.

   recursive step: Suppose T_1 and $T_2 \in \mathcal{E}$ are disjoint. The graph $T_1 \cdot T_2$ formed by taking as root a vertex r not in either T_1, nor T_2, and adding an edge from r to each of the roots of T_1, T_2 (when they are nonempty) is in $\mathcal{E}$.

   Let the set $\mathcal{F}$ be defined as follows:

   basis step: A single vertex r is in $\mathcal{F}$.

   recursive step: Suppose T_1 and $T_2 \in \mathcal{F}$ are disjoint. The graph formed by taking as root a vertex r not in either T_1, nor T_2, and adding an edge from r to each of the roots of T_1, T_2 is in $\mathcal{F}$.

   What differences between the types of trees found in $\mathcal{R}$, $\mathcal{E}$, and $\mathcal{F}$?

10. For the entries of $\mathcal{F}$ defined above, we define a **height** function recursively:

    basis step: If the tree T consists only of a root, then $h(T) = 0$.
    recursive step: For trees T_1 and $T_2 \in \mathcal{F}$, $h(T_1 \cdot T_2) = 1 + \max(h(T_1), h(T_2))$.

11. For the entries of $\mathcal{F}$ defined above, we define the set of **leaves** recursively:

    basis step: If the tree T consists only of a root r, then $L(T) = \{r\}$.
    recursive step: Given trees T_1 and $T_2 \in \mathcal{F}$, $L(T_1 \cdot T_2) = L(T_1) \cup L(T_2)$.

12. For the entries of $\mathcal{F}$ defined above, we define the set of **internal vertices** recursively:

    basis step: If the tree T consists only of a root, then $I(T) = \varnothing$.
    recursive step: Given trees T_1 and $T_2 \in \mathcal{F}$, $I(T_1 \cdot T_2) = \{r\} \cup I(T_1) \cup I(T_2)$, where r is the root of $T_1 \cdot T_2$.

$n(T)$ = # of vertices of T $= |L(T)| + |I(T)|$

Theorem: If $T \in F$, then $n(T) \leq 2^{h(T)+1} - 1$.

$$P(T): \quad n(T) \leq 2^{h(T)+1} - 1.$$

Use structural induction, since the objects involved were defined recursively.

basis step: $P(\cdot)$ holds

basis elements of F

A single-vertex binary tree T has

$n(T) = 1$

$h(T) = 0$

so $n(T) = 1 \overset{?}{\leq} 2^{0+1} - 1$

induction step $\quad P(k) \to P(k+1)$ math induction

$P(1) \wedge P(2) \wedge \ldots \wedge P(k) \to P(k+1)$ strong induction

both assume inputs to P are integers.

So, in structural induction we look at formation of a new tree in recursive step defining F, we assume inputs (trees T_1, T_2) satisfy our claim, then show the new contraction $(T_1 \cdot T_2)$ also satisfies it.

Assume (for the induction step) that our input trees T_1, T_2 have vertex counts and heights satisfying the claim

$$n(T_1) \leq 2^{h(T_1)+1} - 1$$

$$n(T_2) \leq 2^{h(T_2)+1} - 1.$$

Now look at the new tree joining T_1, T_2

$$n(T_1 \cdot T_2) = 1 + n(T_1) + n(T_2)$$

$$\leq \cancel{1} + \left[2^{h(T_1)+1} \cancel{-1}\right] + \left[2^{h(T_2)+1} - 1\right]$$

$$= 2^{h(T_1)+1} + 2^{h(T_2)+1} - 1$$

$$\leq 2 \cdot \max\left(2^{h(T_1)+1}, 2^{h(T_2)+1}\right) - 1$$

$$= 2 \cdot 2^{\underbrace{\max(h(T_1), h(T_2))+1}_{= h(T_1 \cdot T_2)}} - 1$$

$$= 2 \cdot 2^{h(T_1 \cdot T_2)} - 1$$

$$= 2^{h(T_1 \cdot T_2)+1} - 1$$

So $\boxed{n(T_1 \cdot T_2) \leq 2^{h(T_1 \cdot T_2)+1} - 1.}$