CSE 311: Foundations of Computing

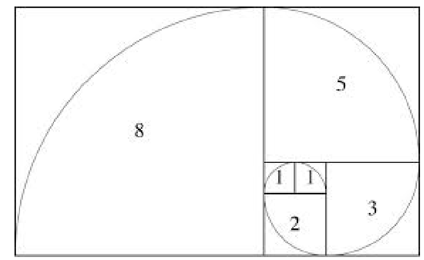
Lecture 19: Structural induction

Image from @Hevesh5 on YouTube

Fibonacci Numbers

$gcd(a, b) = gcd(b, a \% b)$

$$f_0 = 0$$
$$f_1 = 1$$
$$f_n = f_{n-1} + f_{n-2}$$ for all $n \geq 2$



Last lecture: $f_n < 2^n$ for all $n \geq 0$

Similar: $f_n \geq 2^{\frac{n}{2}-1}$ for all $n \geq 2$

Theorem: Suppose that Euclid's Algorithm takes n steps for $\gcd(a, b)$ with $a \geq b > 0$. Then, $a \geq f_{n+1}$.

This implies: $n \leq 1 + 2\log_2 a$

i.e., # of steps $\leq 1 +$ twice the # of bits in a.

Recap: Recursive Definitions of Sets

Recursive definition

- *Basis step:* Some specific elements are in S
- *Recursive step:* Given some existing named elements in S some new objects constructed from these named elements are also in S.
- *Exclusion rule*: Every element in S follows from basis steps and a finite number of recursive steps

Example: The set **Σ^*** of *strings* over the alphabet **Σ** is defined by

- **Basis:** $\varepsilon \in \Sigma^*$ (ε is the empty string)
- **Recursive:** if $w \in \Sigma^*$, $a \in \Sigma$, then $wa \in \Sigma^*$

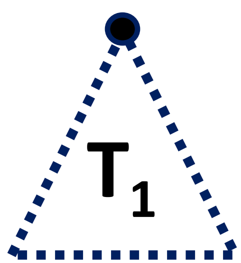
Example of function for recursively defined set: String length

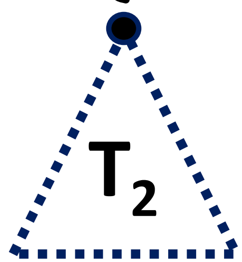
- len(ε) = 0
- len(wa) = 1 + len(w) for $w \in \Sigma^*$, $a \in \Sigma$

Rooted Binary Trees

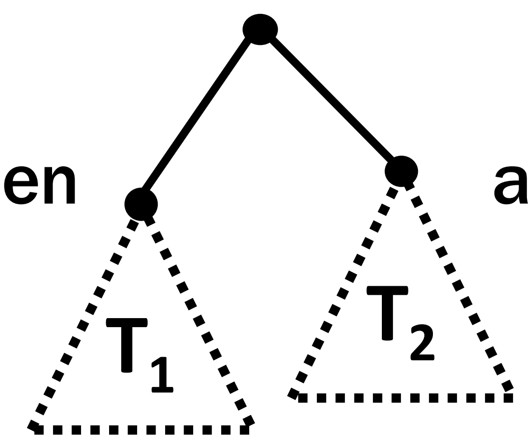
- **Basis:** • is a rooted binary tree
- **Recursive step:**

Tree(L_1, R_1) Tree(L_2, R_2)

If 

and 

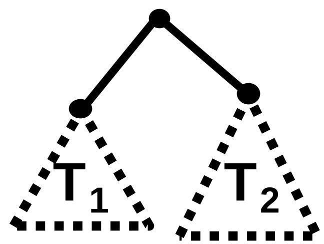
are rooted binary trees,

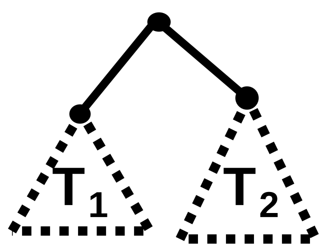
then 

also is a rooted binary tree.

Tree(Tree(L_1, R_1) Tree(L_2, R_2))

Defining Functions on Rooted Binary Trees

- size(•) = 1
- size (

) = 1 + size($\mathbf{T_1}$) + size($\mathbf{T_2}$)
- height(•) = 0
- height (

)=1 + max{height($\mathbf{T_1}$), height($\mathbf{T_2}$)}

Structural Induction

How to prove $\forall\, x \in S, P(x)$ is true:

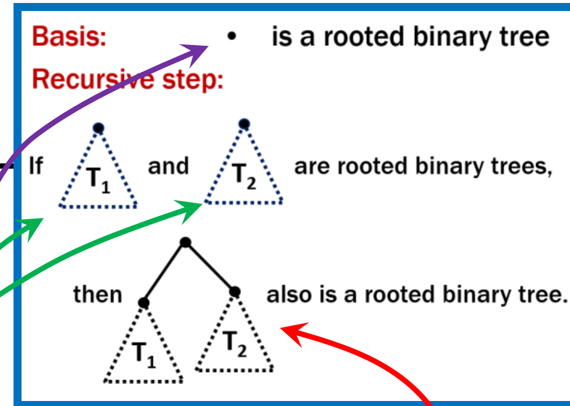
Base Case: Show that $P(u)$ is true for all specific elements u of S mentioned in the *Basis step*

Inductive Hypothesis: Assume that P is true for some arbitrary values of *each* of the existing named elements mentioned in the *Recursive step*

Inductive Step: Prove that $P(w)$ holds for each of the new elements w constructed in the *Recursive step* using the named elements mentioned in the Inductive Hypothesis

Conclude that $\forall\, x \in S, P(x)$

Structural Induction



How to prove $\forall\, x \in S, P(x)$ is true:

Base Case: Show that $P(u)$ is true for all specific elements u of S mentioned in the *Basis step*

Inductive Hypothesis: Assume that P is true for some arbitrary values of *each* of the existing named elements mentioned in the *Recursive step*

Inductive Step: Prove that $P(w)$ holds for each of the new elements w constructed in the *Recursive step* using the named elements mentioned in the Inductive Hypothesis

Conclude that $\forall\, x \in S, P(x)$

Using Structural Induction

- Let S be given by…
  - **Basis:** $6 \in S$; $15 \in S$;
  - **Recursive:** if $x, y \in S$ then $x + y \in S$.

Claim: Every element of S is divisible by 3.

Claim: Every element of S is divisible by 3.

1. Let $P(x)$ be "$3|x$". We prove that $P(x)$ is true for all $x \in S$ by structural induction.

Basis: $6 \in S$; $15 \in S$;

Recursive: if $x, y \in S$ then $x + y \in S$

Claim: Every element of S is divisible by 3.

1. Let $P(x)$ be "$3|x$". We prove that $P(x)$ is true for all $x \in S$ by structural induction.
2. Base Case: $3|6$ and $3|15$ so $P(6)$ and $P(15)$ are true

Basis: $6 \in S$; $15 \in S$;

Recursive: if $x, y \in S$ then $x + y \in S$

Claim: Every element of S is divisible by 3.

1. Let P(x) be "3|x". We prove that P(x) is true for all x ∈ S by structural induction.
2. Base Case: 3|6 and 3|15 so P(6) and P(15) are true
3. Inductive Hypothesis: Suppose that P(x) and P(y) are true for some arbitrary x,y ∈ S
4. Inductive Step: Goal: Show P(x+y)

Basis: $6 \in S$; $15 \in S$;

Recursive: if $x, y \in S$ then $x + y \in S$

Claim: Every element of S is divisible by 3.

1. **Let** P(x) **be** "3|x". **We prove that** P(x) **is true for all** x ∈ S **by structural induction.**
2. **Base Case:** 3|6 **and** 3|15 **so** P(6) **and** P(15) **are true**
3. **Inductive Hypothesis: Suppose that** P(x) **and** P(y) **are true for some arbitrary** x,y ∈ S
4. **Inductive Step:** **Goal: Show** P(x+y)

   Since P(x) **is true,** 3|x **and so** x=3m **for some integer** m **and since** P(y) **is true,** 3|y **and so** y=3n **for some integer** n. **Therefore** x+y=3m+3n=3(m+n) **and thus** 3|(x+y).

   Hence P(x+y) **is true.**
5. **Therefore by induction** 3|x **for all** x ∈ S.

Basis: $6 \in S$; $15 \in S$;

Recursive: if $x, y \in S$ then $x + y \in S$

Structural Induction vs. Ordinary Induction

Ordinary induction is a special case of structural induction:

Recursive definition of $\mathbb{N}$

Basis: $0 \in \mathbb{N}$

Recursive step: If $k \in \mathbb{N}$ then $k + 1 \in \mathbb{N}$

Structural induction follows from ordinary induction:

Define $Q(n)$ to be "for all $x \in S$ that can be constructed in at most n recursive steps, $P(x)$ is true."

Lecture 19 Activity

if $\Sigma = \{0, 1\}$

$\Sigma^* = \{\varepsilon, 0, 1, 00, 01, 10, 11, \ldots\}$

You will be assigned to breakout rooms. Please:

- Introduce yourself
- Choose someone to share their screen, showing this PDF
- Recall that we defined recursively
  - Strings. Basis: $\varepsilon \in \Sigma^*$ Recursive: If $w \in \Sigma^*$ and $a \in \Sigma$ then $wa \in \Sigma^*$
  - Concatenation of strings. Basis: $x \cdot \varepsilon = x$ for Σ^*
    Recursive: $x \cdot wa = (x \cdot w)a$ for $x \in \Sigma^*$, $a \in \Sigma$
- Let $P(y)$ be ``$\text{len}(x \cdot y) = \text{len}(x) + \text{len}(y)$ for all $x \in \Sigma^*$''
- We want to prove $P(y)$ fo all $y \in \Sigma^*$ by structural induction. Please complete the proof:

Base Case: Let $x \in \Sigma^*$ arbitrary. Then $\text{len}(x \cdot \varepsilon) = \text{len}(x) = \text{len}(x) + 0 = \text{len}(x) + \text{len}(\varepsilon)$. Hence $P(\varepsilon)$.

Inductive Hypothesis: Suppose $P(w)$ for an arbitrary $w \in \Sigma^*$.

Inductive Step: ...

Fill out the poll everywhere for Activity Credit!
Go to pollev.com/philipmg and login with your UW identity

Lecture 19 Activity

- Recall that we defined recursively
  - Strings. Basis: $\varepsilon \in \Sigma^*$ Recursive: If $w \in \Sigma^*$ and $a \in \Sigma$ then $wa \in \Sigma^*$
  - Concatenation of strings. Basis: $x \cdot \varepsilon = x$ for Σ^*
    Recursive: $x \cdot wa = (x \cdot w)a$ for $x \in \Sigma^*$, $a \in \Sigma$
- Let $P(y)$ be ``$\text{len}(x \cdot y) = \text{len}(x) + \text{len}(y)$ for all $x \in \Sigma^*$''
- We want to prove $P(y)$ fo all $y \in \Sigma^*$ by structural induction. Please complete the proof:

Base Case: Let $x \in \Sigma^*$ arbitrary. Then $\text{len}(x \cdot \varepsilon) = \text{len}(x) = \text{len}(x) + 0 = \text{len}(x) + \text{len}(\varepsilon)$. Hence $P(\varepsilon)$.

Inductive Hypothesis: Suppose $P(w)$ for an arbitrary $w \in \Sigma^*$.

Inductive Step: Let $a \in \Sigma$. Let $x \in \Sigma^*$. Then $\text{len}(x \bullet wa) = \text{len}((x \bullet w)a)$ by defn of $\bullet$

$= \text{len}(x \bullet w)+1$ by defn of len

$= \text{len}(x)+\text{len}(w)+1$ by I.H.

$= \text{len}(x)+\text{len}(wa)$ by defn of len

Therefore $\text{len}(x \bullet wa)= \text{len}(x)+\text{len}(wa)$ for all $x \in \Sigma^*$, so $P(wa)$ is true.

So, by induction $\text{len}(x \bullet y) = \text{len}(x) + \text{len}(y)$ for all $x,y \in \Sigma^*$

Claim: For every rooted binary tree **T**, $\text{size}(\mathbf{T}) \leq 2^{\text{height}(\mathbf{T}) + 1} - 1$

Claim: For every rooted binary tree **T**, $\text{size}(\mathbf{T}) \le 2^{\text{height}(\mathbf{T}) + 1} - 1$

1. **Let** $P(\mathbf{T})$ **be** "$\text{size}(T) \le 2^{\text{height}(T)+1} - 1$". **We prove** $P(\mathbf{T})$ **for all rooted binary trees T by structural induction.**

Claim: **For every rooted binary tree T,** $\text{size}(\mathbf{T}) \le 2^{\text{height}(\mathbf{T}) + 1} - 1$

1. **Let** $P(\mathbf{T})$ **be** "$\text{size}(T) \le 2^{\text{height}(T)+1}-1$". **We prove** $P(\mathbf{T})$ **for all rooted binary trees T by structural induction.**
2. **Base Case:** $\text{size}(\bullet)=1$, $\text{height}(\bullet)=0$, **and** $2^{0+1}-1=2^1-1=1$ **so** $P(\bullet)$ **is true.**

Claim: **For every rooted binary tree T,** size(**T**) $\leq 2^{\text{height}(\mathbf{T})\,+\,1} - 1$

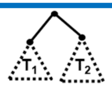
---

1. Let P(**T**) **be** "size(T) $\leq 2^{\text{height}(T)+1}-1$"**. We prove** P(**T**) **for all rooted binary trees T by structural induction.**

2. Base Case: size(•)=1, height(•)=0, **and** $2^{0+1}-1=2^{1}-1=1$ **so** P(•) **is true.**

3. Inductive Hypothesis: Suppose that P(**T_1**) **and** P(**T_2**) **are true for some rooted binary trees T_1 and T_2.**

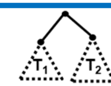
4. Inductive Step:

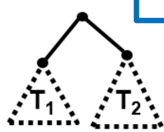
Goal: Prove P(

)**.**

Claim: **For every rooted binary tree T,** size(**T**) $\leq 2^{\text{height}(\mathbf{T})+1} - 1$

---

1. **Let** P(**T**) **be** "size(T) $\leq 2^{\text{height}(T)+1}-1$". **We prove** P(**T**) **for all rooted binary trees T by structural induction.**
2. **Base Case:** size(•)=1, height(•)=0, **and** $2^{0+1}-1=2^1-1=1$ **so** P(•) **is true.**
3. **Inductive Hypothesis: Suppose that** P($\mathbf{T_1}$) **and** P($\mathbf{T_2}$) **are true for some rooted binary trees** $\mathbf{T_1}$ **and** $\mathbf{T_2}$.
4. **Inductive Step:** **Goal: Prove** P().

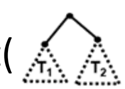
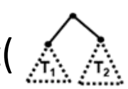
By defn, size() $=1+\text{size}(\mathbf{T_1})+\text{size}(\mathbf{T_2})$

$\leq 1+2^{\text{height}(\mathbf{T_1})+1}-1+2^{\text{height}(\mathbf{T_2})+1}-1$

by IH for $\mathbf{T_1}$ and $\mathbf{T_2}$

$\leq 2^{\text{height}(\mathbf{T_1})+1}+2^{\text{height}(\mathbf{T_2})+1}-1$

$\leq 2(2^{\max(\text{height}(\mathbf{T_1}),\text{height}(\mathbf{T_2}))+1})-1$

$\leq 2(2^{\text{height}(}$  $^{)})-1 \leq 2^{\text{height}(}$  $^{)+1}-1$

which is what we wanted to show.

5. **So, the** P(T) **is true for all rooted bin. trees by structural induction.**