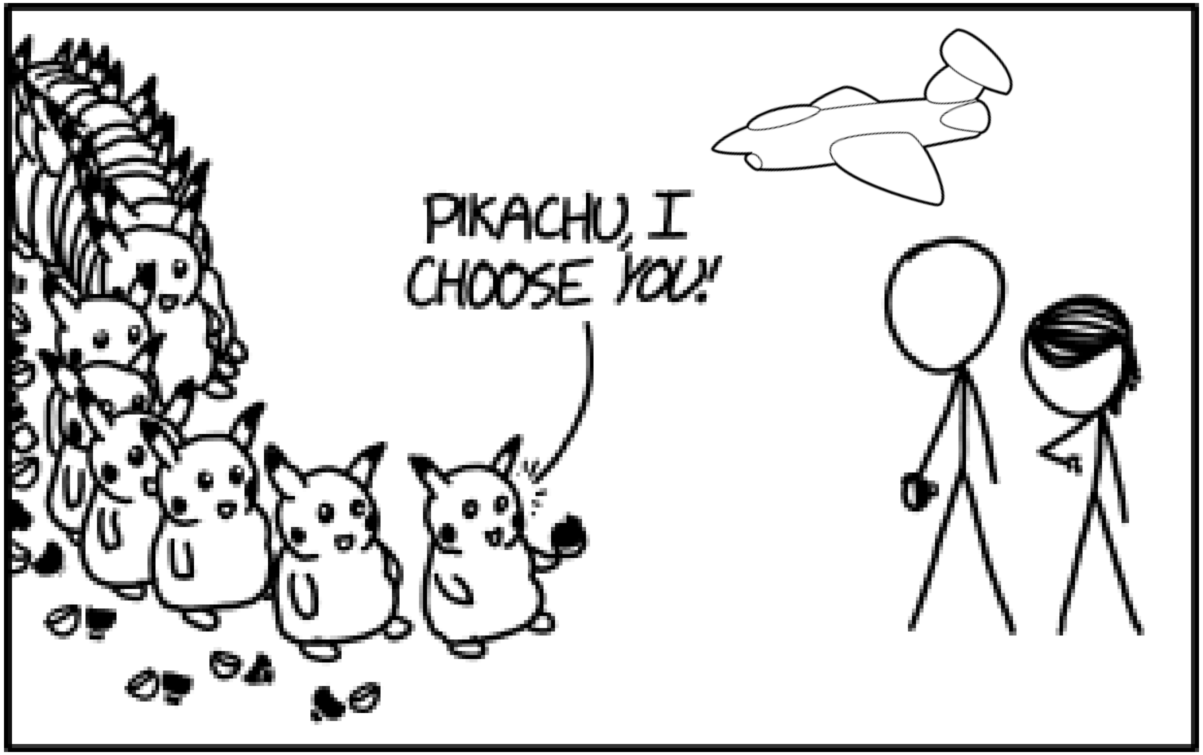
Spring 2015

Lecture 18:
Recursively defined sets and structural induction



Four weeks left: What happens now?

The class speeds up a bit.
Homework problems get more conceptual.

We will cover:

- Recursively defined sets and functions
- Structural induction
- Regular expressions and context free grammars
- Relations and graphs
- Finite state machines and automata
- Turing machines and undecidability

Recursive definition

- Basis step: $0 \in S$
- Recursive step: if $x \in S$, then $x + 2 \in S$
- Exclusion rule: Every element in S follows from basis steps and a finite number of recursive steps

recursive definition of sets

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

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

$S = \{6a + 15b : a, b \geq 0, a, b \in \mathbb{N}\}$

Basis: $[1, 1, 0] \in S, [0, 1, 1] \in S$;

Recursive:

if $[x, y, z] \in S$, $\alpha \in \mathbb{R}$, then $[\alpha x, \alpha y, \alpha z] \in S$

if $[x_1, y_1, z_1], [x_2, y_2, z_2] \in S$

then $[x_1 + x_2, y_1 + y_2, z_1 + z_2] \in S$

~~Powers of 3:~~

Basis: $[0, 0, 1] \in S$ $\quad S = \mathrm{span}((1,1,0),(0,1,1))$

Basis: $1 \in S$ $\quad$ Recursive: $x \in S \Rightarrow 3x \in S$

recursive definitions of sets: general form

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

- An *alphabet* Σ is any finite set of characters.

  e.g. $\Sigma = \{0,1\}$ or $\Sigma = \{A, B, C, \ldots X, Y, Z\}$ or

  $\Sigma =$

1		28	∟	95	_	153	Ö	186	║	219	█
2	☻	29	↔	96	`	154	Ü	187	╗	220	▄
3	♥	30	▲	97-122	a-z	155	¢	188	╝	221	▌
4	♦	31	▼	123	{	156	£	189	╜	222	▐
5	♣	32	(space)	124	|	157	¥	190	╛	223	▀
6	♠	33	!	125	}	158	₧	191	┐	224	α
7	•	34	"	126	~	159	ƒ	192	└	225	ß
8	◘	35	#	127	⌂	160	á	193	┴	226	Γ
9	○	36	$	128	Ç	161	í	194	┬	227	π
10	◙	37	%	129	ü	162	ó	195	├	228	Σ
11	♂	38	&	130	é	163	ú	196	─	229	σ

- 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 $wa \in \Sigma^*$

Palindromes are strings that are the same backwards and forwards. Σ = finite alphabet

Basis:

ε is a palindrome and any $a \in \Sigma$ is a palindrome

Recursive step:

If p is a palindrome then apa is a palindrome for every $a \in \Sigma$.

binary strings such that…

First digit cannot be a 1.

Basis: $0 \in S$, $\varepsilon \in S$

Recursive step: If $x \in S$ and $x \neq \varepsilon$

then $x0 \in S$

$x1 \in S$.

If $x = \varepsilon$ then $x0 \in S$.

* No occurrence of the substring 11.

Basis: $\varepsilon \in S$

Recursive step: If $\overbrace{x_1 \cdots x_k}^{x} \in S$ and $x_k \neq 1$

then $x_1 \cdots x_k 1 \in S$

If $x \in S$ then $x0 \in S$

function definitions on recursively defined sets

$x \bullet y$ Σ^*

Length:

len (ε) = 0;

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

$len(011) = len(01) + 1$
$= len(0) + 1 + 1$
$= len(0) + 2 = len(\varepsilon) + 3$
$= 3.$

Reversal:

$\varepsilon^R = \varepsilon$

$(wa)^R = aw^R$ for $w \in \Sigma^*$, $a \in \Sigma$

$(011)^R = 1(01)^R$
$= 110^R$
$= 110\varepsilon^R = 110\varepsilon = 110$

Concatenation:

$x \bullet \varepsilon = x$ for $x \in \Sigma^*$

$x \bullet wa$ = $(x \bullet w)a$ for $x, w \in \Sigma^*$, $a \in \Sigma$

$011 \bullet 00 = (011 \bullet 0)0$
$= (011 \bullet \varepsilon)00$
$= 01100$

function definitions on recursively defined sets

Number of vowels in a string:

$\Sigma = \{a, b, c, \ldots, z\}$

$\mathcal{V} = \{a, e, i, o, u\}$

vow(x)

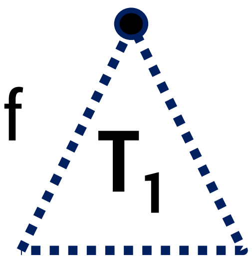
Basis: $\text{vow}(\varepsilon) = 0$

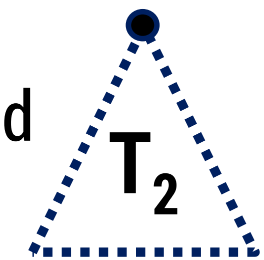
Recursive step: For $w \in \Sigma^*$ and $a \in \Sigma$

$$\text{vow}(wa) = \begin{cases} 1 + \text{vow}(w) & a \in \mathcal{V} \\ \text{vow}(w) & a \notin \mathcal{V} \end{cases}$$

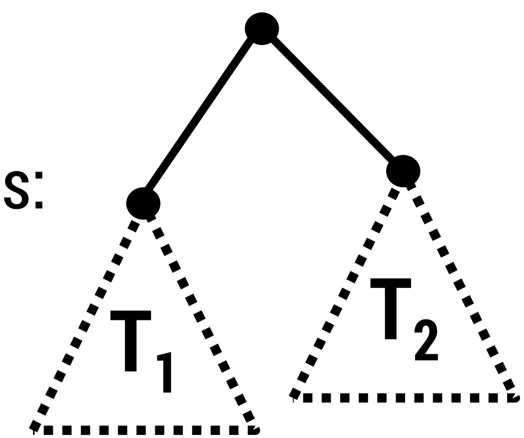
rooted binary trees

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

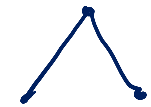
If 

and 

are rooted binary trees,

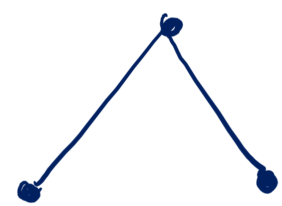
then so is: 

$T_1 = •$

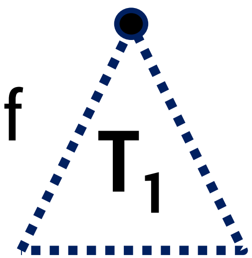
$T_2 =$ 

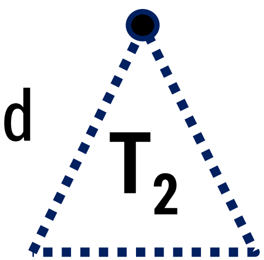
$T_1 = •$

$T_2 = •$

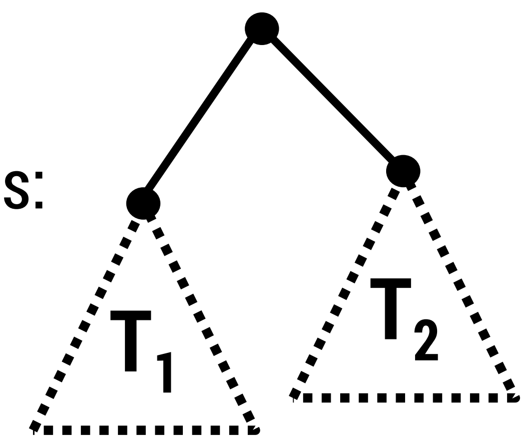


- Basis: • is a rooted binary tree
- Recursive step:

If 

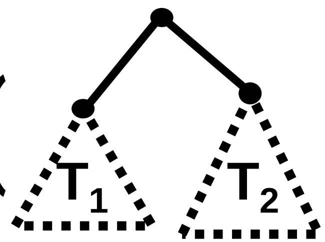
and 

are rooted binary trees,

then so is: 

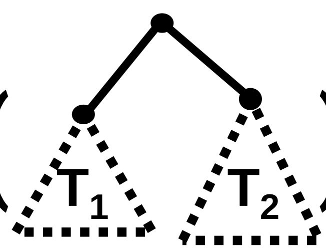
defining a function on rooted binary trees

- size(•) = 1

- size (

) = 1 + size(T_1) + size(T_2)

- height(•) = 0

- height (

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

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*

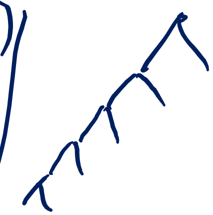
Inductive Hypothesis: Assume that P is true for some arbitrary values of *each* of the existing named elements mentioned in the *Recursive step*

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

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

$\text{size}(T) \geq 2\,\text{height}(T) + 1$



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

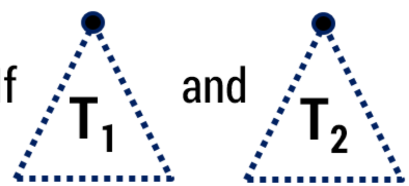
Inductive Hypothes
arbitrary values of
mentioned in the *R*

Inductive Step: Pr
elements w const
named elements m

Conclude that $\forall\, x$

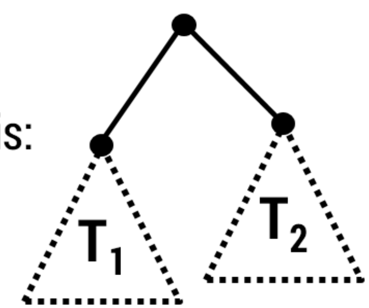
- Basis: • is a rooted binary tree
- Recursive step:

If



are rooted binary trees,

then so is:



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 $k + 1 \in \mathbb{N}$

$\forall k \in \mathbb{N}\ P(k)$

Structural induction follows from ordinary induction:

Let $Q(n)$ be true iff for all $x \in S$ that take n recursive steps to be constructed, $P(x)$ is true.

$\forall x \in S\ P(x)$

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.

$P(x)$ = "x is divisible by 3"

Base case: $6 = 2 \cdot 3 \rightarrow P(6)$

$15 = 5 \cdot 3 \longrightarrow P(15)$

IH: Assume $P(x)$ and $P(y)$ for $x, y \in S$

Goal: $P(x+y)$

$P(x) \Rightarrow x = 3i$ $\Rightarrow x+y = 3(i+j)$

$P(y) \Rightarrow y = 3j$ $\Rightarrow 3 \mid x+y$

$\Rightarrow P(x+y)$.

By str. ind, $\forall x \in S\ P(x)$.