Recursively Defined Structures / Sets

A recursively defined set is a set S defined by

(1) its smallest element (base case)

(2) rules that construct compound elements out of smaller elements. (recursive case)

$$S = \{ x : x \text{ is (1) or } x \text{ follows (2)} \}$$

<u>ex</u> non-negative integers

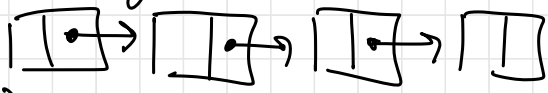
(1) 0

(2) $1 + k$ for a non-negative int k

How do I make 1?

0 is a non-neg. integer by (1)

$1 + 0$ is a non-neg. integer by (2)

<u>ex</u> linked lists 

(1) an empty list ( )

(2) A linked list (x, L) where x is

some data and L is a linked list.

1-element linked list:

$(x, ())$

2-element linked list:

$(x, (y, ()))$

$p \wedge q \Rightarrow r$

$p \vee q \Rightarrow p$

<u>ex</u> well-formed statements of propositional logic over a set of variables X

(1) p, for some $p \in X$

(2a) $p \star q$ where $\star \in \{\wedge, \vee, \Rightarrow, \Leftrightarrow, \oplus\}$ and p, q well-formed statements

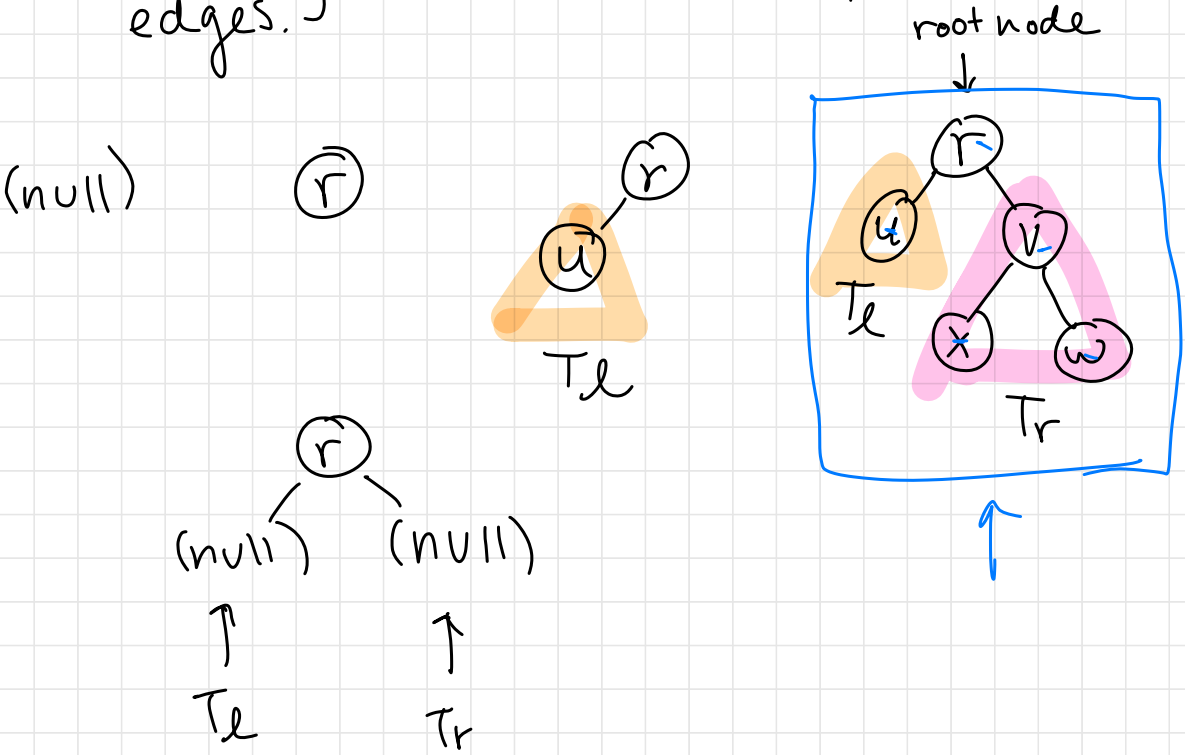
(2b) $\neg p$, p a well-formed statement

$\neg(p \wedge q \Rightarrow r \Leftrightarrow p \vee q \Rightarrow p)$

<u>ex</u> binary trees

(1) null (empty tree)

(2) root node r and T_ℓ, T_r binary trees attached to r with edges.



Terms:
- binary because nodes have 2 children
- edges connect pairs of nodes
- node is a leaf if it has no children (both null)
- node is an internal node if not a leaf

Claim: In any binary tree T,

$$\#\text{leaves}(T) \le \#\text{internal nodes}(T) + 1$$

$$3 \le 2 + 1$$

We could prove w/ structural induction.

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

How:

1) Prove $P(x)$ holds for all base cases of S

2) Prove that if $P(x)$ is T for smaller elements of S, then it is true for larger elements.