# Proofs about Proof Systems

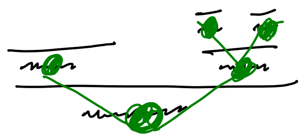
Proposition — statement can be judged as either true or false

$\underbrace{1+3 \Longrightarrow 4}_{\text{true}}$ $\underbrace{\text{True And False} \Longrightarrow \text{True}}_{\text{false}}$

$\underbrace{1+\text{True} \Longrightarrow 2}_{\text{false}}$

Proof: demonstration that a proposition is true

Proof tree: shaped like a tree



$k+0=k$

## Induction

```
let rec sum xs =
  match xs with
    | [] -> 0
    | h::t -> h + sum t
;;
```

* $P(0)$
* $P(n)$ implies $P(n+1)$

$P(n) \equiv$ sum xs returns the arithmetic sum of all integers in xs when xs has length $n$

$P(0)$: sum xs reduces to match w/ xs = [] which reduces to 0; 0 is the additive identity and so sum [] = 0, which is the sum of no integers.

$P(n)$ implies $P(n+1)$:

Want to show that sum xs = arith. sum of all integers in xs when len xs $= n+1$

Know that sum xs = arith. sum of all integers in xs when len xs $= n$

If len xs $= n+1$ and $n \in \mathbb{N}$ then $n+1 > 0$ so h::t case evaluates.

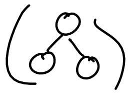
h is an integer in xs, t is rest of list and len t is $n$.

So by inductive hypothesis, sum t is arith sum of all elements in t.

So h + sum t is arith sum of xs, and this is returned.

$\square$

Complete binary tree: empty or node w/ two children of eq height that are also complete binary trees.

$H($  $) = 1$ $\qquad H(\circ) = 0 \qquad H(-) = -1$

Proposition + Proof → Lemma : # nodes in a CBT $= 2^{n+1} - 1$

$P(n) \equiv$ For a CBT of height $n$, # nodes is $2^{n+1} - 1$.

---

By induction on the height of the CBT.

Base case:

$P(-1) \equiv$ For a CBT of height $-1$, # of nodes is $2^{-1+1} - 1 = 1 - 1 = 0$

By defn of empty CBT.

Inductive case:

$P(n)$ implies $P(n+1)$.

WTS that CBT height $n+1$ contains $2^{n+2} - 1$ nodes.

Have by ind hyp that CBT height $n$ contains $2^{n+1} - 1$ nodes.

CBT of height $n+1$ nodes is a node w/ two children that are CBT of height $n$. By ind hyp, CBT height $n+1$ is a node w/ two children, each w/ $2^{n+1} - 1$ nodes. # nodes $= 1 + 2 \cdot (2^{n+1} - 1) = 1 + 2^{n+2} - 2$

$= 2^{n+2} - 1$

$\boxed{\text{BOOL is normalizing.}}$

$e ::= v \mid e \text{ And } e \mid e \text{ Or } e \mid \text{Not } e$

$v ::= \text{True} \mid \text{False}$

$$\frac{}{v \Rightarrow v} \qquad \frac{e_1 \Rightarrow v_1 \quad e_2 \Rightarrow v_2 \quad v \text{ is log and of } v_1 \text{ and } v_2}{e_1 \text{ And } e_2 \Rightarrow v}$$

In BOOL, $\forall e. \exists v. \; e \Rightarrow v$.

$P(n) \equiv$ For all $e$ of height $n$, $\exists v.\; e \Rightarrow v$.

By induction on height of $e$.

Base case: For all $e$ of height 0, $\exists v.\; e \Rightarrow v$.
If $e$ has height 0, then $e = \text{True}$ or $e = \text{False}$.
In both cases, Value rule gives $e \Rightarrow v$ where $e = v$.

Inductive case: WTS for all $e$ of height $n+1$, $\exists v.\; e \Rightarrow v$.

3 cases: $e = e_1 \text{ And } e_2$ or $e = e_1 \text{ Or } e_2$ or $e = \text{Not } e_1$.

Case 1: $e_1 \text{ And } e_2$. b/c $e$ has height $n+1$, $e_1$ and $e_2$ both have height $\leq n$.
By strong induction, we have $e_1 \Rightarrow v_1$ and $e_2 \Rightarrow v_2$.
b/c logical and" is defined in terms of all pairs of $v_1$ and $v_2$,
And rule gives us $e \Rightarrow v$ for $v =$ "logical and" of $v_1$ and $v_2$.