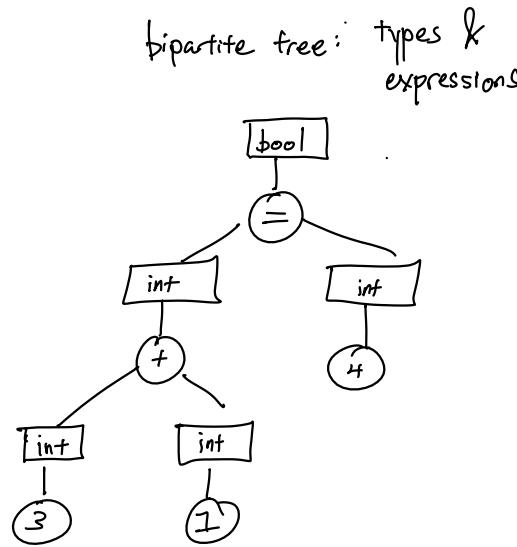
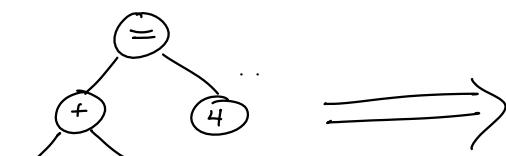


Type System

- What is a type system?
 - Organizes values into "types": ex, "1" is a string; 1 is an int
 - Rules about how types are used
 - Rules about how types are inferred
- What is a type?
 - A type is a set of values.
 - bool = { true, false }
 - int = { 0, 1, -1, ... }
- What is a set?
 - A collection of unique values.

Using Static Typing

$$3 + 1 = 4$$



$$T(3) = \text{int}$$

A diagram showing the type $T(3)$ as a box labeled int with a value 3 below it.

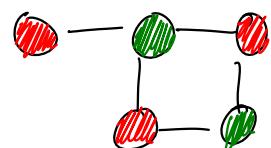
$$T(\text{false}) = \text{bool}$$

A diagram showing the type $T(\text{false})$ as a box labeled bool with a value false below it.

$$T(\text{after } e) = \text{int}$$

A diagram showing the type $T(\text{after } e)$ as a box labeled int with a value e below it. Below this is another int node with a value e below it, indicating a recursive or self-referential type structure.

if $T(e) = \text{int}$



$$T(3) = \boxed{\text{int}} \\ \boxed{3}$$

$$T(\text{false}) = \boxed{\text{bool}} \\ \boxed{\text{false}}$$

$$T(\text{after } e) = \boxed{\text{int}} \\ \boxed{\text{after}} \\ \boxed{e} \\ \boxed{\text{int}}$$

if $T(e) = \boxed{\text{int}}$

$$T(\text{if } e_1 \text{ then } e_2 \text{ else } e_3) = \boxed{\text{z}} \\ \boxed{\text{if}} \\ \boxed{e_1} \\ \boxed{e_2} \quad \boxed{e_3} \\ \boxed{\text{bool}}$$

$T(e_1) = \boxed{\text{bool}}$
 $T(e_2) = \boxed{\text{z}}$
 $T(e_3) = \boxed{\text{z}}$

$$T(\{x \mapsto z, \dots\}, \otimes) = \boxed{\text{z}} \\ \boxed{x}$$

$$\text{compile_expression}(\text{env}, \boxed{+} \boxed{x} \boxed{y}) = \boxed{\text{z}}$$

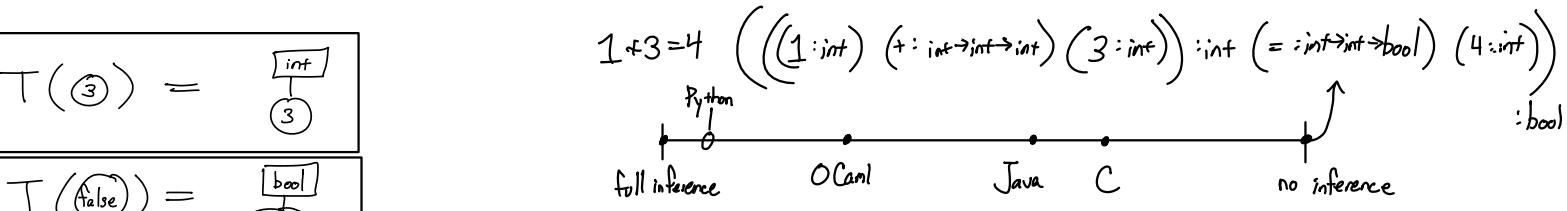
$e_1 \text{ check int}$
 $\text{store } e_1$
 $e_2 \text{ check int}$
 add row, ...

$$\text{compile_expression}(\text{env}, \boxed{+} \boxed{1} \boxed{2}) = \boxed{\text{int}}$$

Same code but without type checks

In compilers, type systems are for:

1. detecting errors
2. proving facts that the compiler can use



$$T(\boxed{+} \boxed{e_1} \boxed{e_2}) = \boxed{\text{int}} \quad \text{if } T(e_1) = \boxed{\text{int}}$$

and $T(e_2) = \boxed{\text{int}}$

$$T(M, \boxed{\text{let}} \boxed{e_1} \boxed{e_2}) = \boxed{\text{z}_2} \quad T(\boxed{\text{z}_1}) = \boxed{\text{z}_1}$$

\times

$$\boxed{\text{let}} \boxed{e_1} \boxed{e_2} \quad \boxed{z_1} \quad \boxed{z_2}$$

$e_1 \quad e_2$

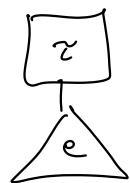
$$T(M \cup \{x \mapsto z_1\}, \boxed{\text{let}} \boxed{e_1}) = \boxed{\text{z}_1}$$

$$T(\boxed{\text{z}_1}) = \boxed{\text{bool}} \quad \text{if } b \text{ then } 4$$

else $(1, 2)$

$$T(\boxed{\text{z}_2}) = \boxed{\text{z}}$$

$$T(\boxed{\text{z}_3}) = \boxed{\text{z}}$$



means that when you run e, you get a value of z
 (or an error, or if runs forever)