Meeting 10: Operational Semantics

RECALL:

What is the purpose of a language specification?
"To give a clear idea of what operations are supposed to do" or "evaluate to"

What is the difference between syntax and semantics?
Syntax = "form of string"

Grammar = BNF = "what you can write down in the language"

Semantics = "meaning"
Operational Semantics = Judges

("interpreter in "math" judgments"

Evaluation Judgment Form

"is a relation between an expression e and its value v and an value env E

E ⊨ e  iff  v  In env E, expression e evaluates to value v
Define judgment forms via inference rules

\[ e ::= n \mid e_1 + e_2 \mid b \mid e_1 \& e_2 \]

\[ v ::= n \mid b \]

Eval Val

\[ E \vdash v \Downarrow v \]

Eval Plus

\[ E \vdash e_1 \Downarrow n_1 \quad E \vdash e_2 \Downarrow n_2 \]

\[ E \vdash e_1 + e_2 \Downarrow n_1 + n_2 \]

Bad rule (Incomplete)

\[ E \vdash n_1 + n_2 \Downarrow n_1 + n_2 \]

Binary (Plus, \( e_1, e_2 \))

Plus is the meta-level plus ("math plus")

3 3+4 3+4+5

\[ \text{x} \]
Work out evaluating \&\&
- both arguments must evaluate boolean
- short-circuiting \&\&
- "right-to-left"

\[
e_1 \&\& e_2
\]

\[
\frac{E e_1 \equiv n_1}{E e_1 \&\& e_2 \equiv n_1 \& n_2}
\]

Eval And False

\[
\frac{E e_2 \equiv \text{false}}{E e_1 \&\& e_2 \equiv \text{false}}
\]

Eval And True

\[
\frac{E e_1 \equiv \text{true} \land E e_2 \equiv \text{true}}{E e_1 \&\& e_2 \equiv \text{true} \land b_1}
\]

\[
(\text{true} \&\& \text{true}) \&\& (\text{false} \&\& \text{true})
\]
Constructing a derivation = test execution

AST & &
   & &
   true true false

Eval Val
   Eval And
     Eval And False
     Eval And False
     true & false & false
     true & false & false
     true & false & false
     true & false & false

Eval Val
   Eval And False
     Eval And False
     Eval And False
     Eval And False
     Eval And False

Eval Val
   Eval And False
     Eval And False
     Eval And False
     Eval And False
     Eval And False
     Eval And False
     Eval And False
((true && true) && true) && false

&&

false

true

true && true