Small-step

\[ e \rightarrow e' \]
\[ b \rightarrow b' \]
\[ \langle c,\sigma \rangle \rightarrow \langle c',\sigma' \rangle \]

\[ \frac{\text{ Winskel!}}{\langle e, \sigma \rangle} \rightarrow \langle e', \sigma' \rangle \]

- e steps to e' in state \( \sigma' \)
- boolean exp b steps to b' in state \( \sigma' \)
- command c in state \( \sigma \)
- steps to command \( c' \) in state \( \sigma' \)
operational semantics
- define meaning of program in terms of how it executes

big step op sem
- relation between program and its final value

="("natural-style op sem ")"
small step $\quad \equiv \quad$ sem.

- expose intermediate states
- "one-step of eval"

\[ (\text{SOS}) = \quad \text{structural sp.} \quad \text{sem.} \]

---

Definition style - "how to define"

structural-style (big-step)

- "define in terms of the structure of expression"
contextual

redexes, contexts, local reduction rules, global reduction rules

\[ <c, o> \rightarrow <c', o'> \]

\[ <r, o> \rightarrow <e, o'> \]

\[ (H \sqcup r) \rightarrow (H \sqcup e, o') \]
store-based

some notion state
\[
\begin{align*}
\text{actual reduction} & \quad \Rightarrow \quad \text{instructions} \\
\text{"search rules"}
\end{align*}
\]