Relation : a subset of a product of sets

$$\mathbb{Z} \times \mathbb{S} = \text{set of pairs of integers and strings}$$

$$\mathbb{Z} \times \mathbb{Z} \times \mathbb{Z} \qquad \square \star \square = \square$$

$$\begin{array}{ccccc} 1 & \star & 1 & = & 2 \\ 2 & \star & 4 & = & 6 \\ 3 & \star & 5 & = & 8 \\ 1 & \star & 1 & = & 8 \end{array}$$

Graph

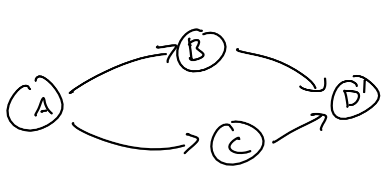
$\langle V, E \rangle$ where E consists of $\langle V, V \rangle$

Grammars : each metavariable represents a different instance regardless of labeling (subscript, etc.)

Rules : same-labeled metavariables are the same value.

Given a graph G, we define a relation $\underline{G} \vdash \underline{V} \rightsquigarrow \underline{V}$.

$$\text{Reflexivity} \ \frac{v \in V}{\langle V, E \rangle \vdash v \rightsquigarrow v} \qquad \text{Neighbor} \ \frac{\langle v_1, v_2 \rangle \in E \quad \langle V, E \rangle \vdash v_2 \rightsquigarrow v_3}{\langle V, E \rangle \vdash v_1 \rightsquigarrow v_3}$$



$$G_0 = \langle \underbrace{\{A, B, C, D\}}_{V_0}, \underbrace{\{\langle A,B \rangle, \langle A,C \rangle, \langle B,D \rangle, \langle C,D \rangle\}}_{E_0} \rangle$$

$$\text{Neighbor} \ \frac{\langle A,B \rangle \in E_0 \quad \text{Neighbor} \ \dfrac{\langle B,D \rangle \in E_0 \quad \text{Reflx} \ \dfrac{}{G_0 \vdash D \rightsquigarrow D}}{G_0 \vdash B \rightsquigarrow D}}{G_0 \vdash A \rightsquigarrow D}$$

Functional dependency : in a relation, some combination of places determines the value in another combination of places.

$$\boxed{1} + \boxed{1} = \boxed{2}$$

$$\square < \square$$

$$\begin{array}{ccc} 2 & < & 4 \\ 3 & < & 4 \\ 2 & < & 3 \end{array}$$

$$\text{height}(\triangle) = \square$$

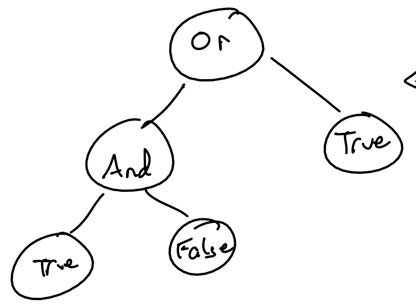
$$\max(\triangle) = \square$$

EBNF grammar

if (true) { x=5; }
if true {) = x5; } (

concrete syntax ⟶ $e ::= e \text{ And } e \mid e \text{ Or } e \mid \text{Not } e \mid \text{True} \mid \text{False} \mid (e)$

(True And False) Or True



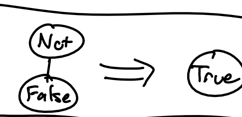
⟵ abstract syntax

Boolean Language Evaluation Relation

$\square \Rightarrow \square$

~~True And False ⇒ False~~

$$\frac{}{\text{Not False} \Rightarrow \text{True}}$$



$$\frac{}{\text{True} \Rightarrow \text{True}} \qquad \frac{}{\text{False} \Rightarrow \text{False}} \qquad \frac{}{\text{Not (Not } e) \Rightarrow e}$$

$$\frac{e \Rightarrow \text{True}}{\text{Not } e \Rightarrow \text{False}} \qquad \frac{e \Rightarrow \text{False}}{\text{Not } e \Rightarrow \text{True}} \qquad \frac{\dfrac{\dfrac{}{\text{True} \Rightarrow \text{True}}}{\text{Not True} \Rightarrow \text{False}}}{\text{Not (Not True)} \Rightarrow \text{True}}$$

$$\frac{e_1 \Rightarrow \text{True} \quad e_2 \Rightarrow \text{True}}{e_1 \text{ And } e_2 \Rightarrow \text{True}} \qquad \frac{e_1 \Rightarrow v \quad e_2 \Rightarrow \text{False}}{e_1 \text{ And } e_2 \Rightarrow \text{False}}$$

$$\frac{e_1 \Rightarrow \text{False} \quad e_2 \Rightarrow v}{e_1 \text{ And } e_2 \Rightarrow \text{False}}$$

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

$e \Rightarrow v$

Operational Semantics

Languages

- Deterministic iff $e \Rightarrow v_1$ and $e \Rightarrow v_2$ implies $v_1 = v_2$
- Normalizing iff $\forall e. \exists v. \ e \Rightarrow v$

Normalizing implies not Turing-complete

- Convergence on an expr: given e, e <u>converges</u> iff $\exists v. \ e \Rightarrow v$
- Divergence: opposite

Expressions