B: Language of Boolean Expressions
Equational Total Function Semantics
CPSC 509: Programming Language Principles

Ronald Garcia

3 February 2022
(**Time Stamp**: 11:05, Monday 11th March, 2024)

Syntax

Presented as BNF

$$t \in \textsc{Term}, \quad v \in \textsc{Value},$$
$$t \quad ::= \quad \mathsf{true} \mid \mathsf{false} \mid \mathsf{if}\ t\ \mathsf{then}\ t\ \mathsf{else}\ t$$
$$v \quad ::= \quad \mathsf{true} \mid \mathsf{false}$$

Equivalent Presentation as Inductive Definitions

$$a \in \textsc{Atom},$$
$$\mathsf{true}, \mathsf{false}, \mathsf{if} \in \textsc{Atom}, \quad \mathsf{true} \neq \mathsf{false}, \quad \mathsf{true} \neq \mathsf{if\text{-}then\text{-}else}, \quad \mathsf{false} \neq \mathsf{if\text{-}then\text{-}else}$$
$$r \in \textsc{Tree}[\textsc{Atom}]$$

$\boxed{\textsc{Term} \subseteq \textsc{Tree}[\textsc{Atom}]}$

$$\frac{}{\mathsf{true} \in \textsc{Term}}\ (\text{rtrue}) \qquad \frac{}{\mathsf{false} \in \textsc{Term}}\ (\text{rfalse}) \qquad \frac{r_1 \in \textsc{Term} \quad r_2 \in \textsc{Term} \quad r_3 \in \textsc{Term}}{\mathsf{if}\ r_1\ \mathsf{then}\ r_2\ \mathsf{else}\ r_3 \in \textsc{Term}}\ (\text{rif})$$

$$\mathcal{R}_{\textsc{Term}} := (\text{rtrue}) \cup (\text{rfalse}) \cup (\text{rif})$$
$$\textsc{Term} := \{\, r \in \textsc{Tree}[\textsc{Atom}] \mid \exists \mathcal{D} \in \textsc{Deriv}[\mathcal{R}_{\textsc{Term}}].\, \mathcal{D} :: r \,\}$$

$\boxed{\textsc{Value} \subseteq \textsc{Tree}[\textsc{Atom}]}$

$$\frac{}{\mathsf{true} \in \textsc{Value}}\ (\text{vtrue}) \qquad \frac{}{\mathsf{false} \in \textsc{Value}}\ (\text{vfalse})$$

$$\mathcal{R}_{\textsc{Value}} := (\text{vtrue}) \cup (\text{vfalse})$$
$$\textsc{Value} := \{\, r \in \textsc{Tree}[\textsc{Atom}] \mid \exists \mathcal{D} \in \textsc{Deriv}[\mathcal{R}_{\textsc{Value}}].\, \mathcal{D} :: r \,\}$$

Total-Function Semantics

$$\textsc{Pgm} = \textsc{Term}, \quad \textsc{Obs} = \textsc{Value}$$
$$eval : \textsc{Pgm} \to \textsc{Obs}$$
$$eval(\textsf{true}) = \textsf{true}$$
$$eval(\textsf{false}) = \textsf{false}$$
$$eval(\textsf{if}\ t_1\ \textsf{then}\ t_2\ \textsf{else}\ t_3) = eval(t_2) \text{ if } eval(t_1) = \textsf{true}$$
$$eval(\textsf{if}\ t_1\ \textsf{then}\ t_2\ \textsf{else}\ t_3) = eval(t_3) \text{ if } eval(t_1) = \textsf{false}$$

Equivalent Presentation as a Definite Description

$$\begin{aligned} eval \equiv\ & \iota F \in \textsc{Pgm} \to \textsc{Obs}. \\ & F(\textsf{true}) = \textsf{true} \wedge \\ & F(\textsf{false}) = \textsf{false} \wedge \\ & \Big(\forall t_1, t_2, t_3 \in \textsc{Term}.\ F(t_1) = \textsf{true} \Rightarrow F(\textsf{if}\ t_1\ \textsf{then}\ t_2\ \textsf{else}\ t_3) = F(t_2)\Big) \wedge \\ & \Big(\forall t_1, t_2, t_3 \in \textsc{Term}.\ F(t_1) = \textsf{false} \Rightarrow F(\textsf{if}\ t_1\ \textsf{then}\ t_2\ \textsf{else}\ t_3) = F(t_3)\Big) \end{aligned}$$

Reasoning Principles

Proposition 1 (Forward-Reasoning for $\textsc{Term} \subseteq \textsc{Tree}[\textsc{Atom}]$)**.**

(rtrue) $\textit{true} \in \textsc{Term}$;

(rfalse) $\textit{false} \in \textsc{Term}$;

(rif) $\forall r_1, r_2, r_3 \in \textsc{Tree}[\textsc{Atom}].\ r_1, r_2, r_3 \in \textsc{Term} \Rightarrow \textit{if}\ r_1\ \textit{then}\ r_2\ \textit{else}\ r_3\ \textit{in}\ \textsc{Term}$.

Proposition 2 (Backward-Reasoning for $\textsc{Term} \subseteq \textsc{Tree}[\textsc{Atom}]$)**.**
$\forall r \in \textsc{Tree}[\textsc{Atom}].\ r \in \textsc{Term} \Rightarrow$
$r = \textit{true} \vee r = \textit{false} \vee (\exists r_1, r_2, r_3 \in \textsc{Tree}[\textsc{Atom}].\ r_1, r_2, r_3 \in \textsc{Term} \wedge r = \textit{if}\ r_1\ \textit{then}\ r_2\ \textit{else}\ r_3\ \textit{in}\ \textsc{Term})$.

Proposition 3 (Backward-Reasoning for $\textsc{Term} \subseteq \textsc{Tree}[\textsc{Atom}]$, Distinguishing $\textsc{Term}$)**.**

(inv-t-true) $\textit{true} \in \textsc{Term} \Rightarrow \top$;

(inv-t-false) $\textit{false} \in \textsc{Term} \Rightarrow \top$;

(inv-t-if) $\forall r_1, r_2, r_3 \in \textsc{Tree}[\textsc{Atom}].\ \textit{if}\ r_1\ \textit{then}\ r_2\ \textit{else}\ r_3\ \textit{in}\ \textsc{Term} \Rightarrow r_1, r_2, r_3 \in \textsc{Term}$.

Proposition 4 (Principle of Derivation Induction for $\textsc{Term} \subseteq \textsc{Tree}[\textsc{Atom}]$)**.**
Let Φ be a predicate on derivations $\mathcal{D} \in \textsc{Deriv}[\mathcal{R}_{\textsc{Term}}]$. Then $\Phi(\mathcal{D})$ holds for all derivations $\mathcal{D}$ if:

1. $\Phi\left(\dfrac{}{\textit{true} \in \textsc{Term}}\ \textit{(rtrue)}\right)$ *holds;*
2. $\Phi\left(\dfrac{}{\textit{false} \in \textsc{Term}}\ \textit{(rfalse)}\right)$ *holds;*
3. $\forall \mathcal{D}_1, D_2, D_3 \in \textsc{Deriv}, r_1, r_2, r_3 \in \textsc{Tree}[\textsc{Atom}].$
   $\mathcal{D}_1 :: r_1 \wedge \mathcal{D}_2 :: r_2 \wedge \mathcal{D}_3 :: r_3 \wedge \Phi(\mathcal{D}_1) \wedge \Phi(\mathcal{D}_2) \wedge \Phi(\mathcal{D}_3) \Rightarrow$
   $$\Phi\left(\dfrac{\overset{\mathcal{D}_1}{r_1 \in \textsc{Term}} \quad \overset{\mathcal{D}_2}{r_2 \in \textsc{Term}} \quad \overset{\mathcal{D}_3}{r_3 \in \textsc{Term}}}{\textit{if}\ r_1\ \textit{then}\ r_2\ \textit{else}\ r_3 \in \textsc{Term}}\ \textit{(rif)}\right).$$

Proposition 5 (Principle of Rule Induction for $\textsc{Term} \subseteq \textsc{Tree}[\textsc{Atom}]$)**.**
Let Φ be a predicate on $\textsc{Tree}[\textsc{Atom}]$. Then $\Phi(t)$ holds for all $t \in \textsc{Term}$ if:

1. *$\Phi($true$)$ holds;*
2. *$\Phi($false$)$ holds;*
3. *For all $r_1, r_2, r_3 \in \textsc{Tree}[\textsc{Atom}]$, if $\Phi(r_1)$, $\Phi(r_2)$, and $\Phi(r_3)$ hold then $\Phi($if r_1 then r_2 else $r_3)$ holds.*

Proposition 6 (Principle of Definition by Recursion for $t \in \textsc{Term}$)**.** *Let S be some set and $s_t, s_f \in S$ be two of its elements and*

$$H_{if} : S \times S \times S \to S$$

be a function on S. Then there exists a unique function

$$F : \textsc{Term} \to S$$

such that

1. $F(\textsf{true}) = s_t$*;*
2. $F(\textsf{false}) = s_f$*;*
3. $F(\textsf{if}\ t_1\ \textsf{then}\ t_2\ \textsf{else}\ t_3) = H_{if}(F(t_1), F(t_2), F(t_3))$*.*

Proposition 7 (Forward-Reasoning for $\textsc{Value} \subseteq \textsc{Tree}[\textsc{Atom}]$)**.**

(vtrue) $\textsf{true} \in \textsc{Value}$*;*

(vfalse) $\textsf{false} \in \textsc{Value}$*.*

Proposition 8 (Backward-Reasoning for $\textsc{Value} \subseteq \textsc{Tree}[\textsc{Atom}]$)**.** $\forall r \in \textsc{Tree}[\textsc{Atom}].\, r \in \textsc{Value} \Rightarrow r = \textsf{true} \lor r = \textsf{false}$*.*

Proposition 9 (Principle of Derivation Induction for $\textsc{Value} \subseteq \textsc{Tree}[\textsc{Atom}]$)**.**
Let Φ be a predicate on derivations $\mathcal{D} \in \textsc{Deriv}[\mathcal{R}_{\textsc{Value}}]$. Then $\Phi(\mathcal{D})$ holds for all derivations $\mathcal{D}$ if:

1. $\Phi\left(\dfrac{}{\textsf{true} \in \textsc{Value}}\ \textit{(vtrue)}\right)$ *holds;*
2. $\Phi\left(\dfrac{}{\textsf{false} \in \textsc{Value}}\ \textit{(vfalse)}\right)$ *holds.*

Proposition 10 (Principle of Rule Induction for $\textsc{Value} \subseteq \textsc{Tree}[\textsc{Atom}]$)**.**
Let Φ be a predicate on $\textsc{Tree}[\textsc{Atom}]$. Then $\Phi(r)$ holds for all $v \in \textsc{Value}$ if:

1. $\Phi(\textsf{true})$ *holds;*
2. $\Phi(\textsf{false})$ *holds.*