

Cheat Sheet

CS231

1 Language of Booleans and Integers

1.1 Syntax

$t ::= \text{true} \mid \text{false} \mid \text{if } t \text{ then } t \text{ else } t$
 $\quad \mid n \mid t + t \mid t > t$
 $n ::= \text{integer constant}$
 $v ::= \text{true} \mid \text{false} \mid n$
 $T ::= \text{Bool} \mid \text{Int}$

1.2 Small-Step Operational Semantics

$$\frac{}{\text{if true then } t_2 \text{ else } t_3 \longrightarrow t_2} \quad (\text{E-IFTRUE})$$

$$\frac{}{\text{if false then } t_2 \text{ else } t_3 \longrightarrow t_3} \quad (\text{E-IFFALSE})$$

$$\frac{t_1 \longrightarrow t'_1}{\text{if } t_1 \text{ then } t_2 \text{ else } t_3 \longrightarrow \text{if } t'_1 \text{ then } t_2 \text{ else } t_3} \quad (\text{E-IF})$$

$$\frac{t_1 \longrightarrow t'_1}{t_1 + t_2 \longrightarrow t'_1 + t_2} \quad (\text{E-PLUS1})$$

$$\frac{t_2 \longrightarrow t'_2}{v_1 + t_2 \longrightarrow v_1 + t'_2} \quad (\text{E-PLUS2})$$

$$\frac{n = n_1 \llbracket + \rrbracket n_2}{n_1 + n_2 \longrightarrow n} \quad (\text{E-PLUSRED})$$

$$\frac{t_1 \longrightarrow t'_1}{t_1 > t_2 \longrightarrow t'_1 > t_2} \quad (\text{E-GT1})$$

$$\frac{t_2 \longrightarrow t'_2}{v_1 > t_2 \longrightarrow v_1 > t'_2} \quad (\text{E-GT2})$$

$$\frac{v = n_1 \llbracket > \rrbracket n_2}{n_1 > n_2 \longrightarrow v} \quad (\text{E-GTRED})$$

1.3 Static Type System

$$\frac{}{\text{true} : \text{Bool}} \quad (\text{T-TRUE}) \qquad \frac{}{\text{false} : \text{Bool}} \quad (\text{T-FALSE})$$

$$\frac{t_1 : \text{Bool} \quad t_2 : T \quad t_3 : T}{\text{if } t_1 \text{ then } t_2 \text{ else } t_3 : T} \quad (\text{T-IF})$$

$$\frac{}{n : \text{Int}} \quad (\text{T-NUM})$$

$$\frac{t_1 : \text{Int} \quad t_2 : \text{Int}}{t_1 + t_2 : \text{Int}} \quad (\text{T-PLUS})$$

$$\frac{t_1 : \text{Int} \quad t_2 : \text{Int}}{t_1 > t_2 : \text{Bool}} \quad (\text{T-GT})$$

2 Simply-Typed Lambda Calculus

2.1 Syntax

$t ::= x \mid \text{function } x:T \rightarrow t \mid t \ t$
 $v ::= \text{function } x:T \rightarrow t$
 $T ::= T_1 \rightarrow T_2$

2.2 Substitution

$[x \mapsto v]x = v$
 $[x \mapsto v]x' = x', \text{ where } x \neq x'$
 $[x \mapsto v]\text{function } x:T \rightarrow t_0 = \text{function } x:T \rightarrow t_0$
 $[x \mapsto v]\text{function } x_0:T \rightarrow t_0 = \text{function } x_0:T \rightarrow [x \mapsto v]t_0, \text{ where } x \neq x_0$
 $[x \mapsto v]t_1 \ t_2 = [x \mapsto v]t_1 \ [x \mapsto v]t_2$

2.3 Small-Step Operational Semantics

$$\frac{}{((\text{function } x:T \rightarrow t) \ v) \longrightarrow [x \mapsto v]t} \quad (\text{E-APPBETA})$$

$$\frac{t_1 \longrightarrow t'_1}{t_1 \ t_2 \longrightarrow t'_1 \ t_2} \quad (\text{E-APP1})$$

$$\frac{t_2 \longrightarrow t'_2}{v_1 \ t_2 \longrightarrow v_1 \ t'_2} \quad (\text{E-APP2})$$

2.4 Static Type System

Γ is a finite function from variable names to types.

$$\frac{\Gamma(x) = T}{\Gamma \vdash x : T} \quad (\text{T-VAR})$$

$$\frac{\Gamma, x:T_1 \vdash t : T_2}{\Gamma \vdash \text{function } x:T_1 \rightarrow t : T_1 \rightarrow T_2} \quad (\text{T-FUN})$$

$$\frac{\Gamma \vdash t_1 : T_2 \rightarrow T \quad \Gamma \vdash t_2 : T_2}{\Gamma \vdash t_1 \ t_2 : T} \quad (\text{T-APP})$$

3 Extensions

We augment our language with a unit value, pairs, tagged unions, let, letrec.

3.1 Syntax

$t ::= () \mid (t, t) \mid \text{fst } t \mid \text{snd } t$
 $\quad \mid \text{left } t \mid \text{right } t \mid (\text{match } t \text{ with left } x \rightarrow t \mid \text{right } x \rightarrow t)$
 $\quad \mid \text{let } x=t \text{ in } t \mid \text{letrec } x=v \text{ in } t$
 $v ::= () \mid (v, v) \mid \text{left } v \mid \text{right } v$
 $T ::= \text{True} \mid \text{False} \mid T \wedge T \mid T \vee T$

3.2 Small-Step Operational Semantics

$\frac{t_1 \rightarrow t'_1}{(t_1, t_2) \rightarrow (t'_1, t_2)} \quad (\text{E-PAIR1})$	$\frac{t_2 \rightarrow t'_2}{(v_1, t_2) \rightarrow (v_1, t'_2)} \quad (\text{E-PAIR2})$
$\frac{t \rightarrow t'}{\text{fst } t \rightarrow \text{fst } t'} \quad (\text{E-FST})$	$\frac{}{\text{fst } (v_1, v_2) \rightarrow v_1} \quad (\text{E-FSTRED})$
$\frac{t \rightarrow t'}{\text{snd } t \rightarrow \text{snd } t'} \quad (\text{E-SND})$	$\frac{}{\text{snd } (v_1, v_2) \rightarrow v_2} \quad (\text{E-SNDRED})$
$\frac{t \rightarrow t'}{\text{left } t \rightarrow \text{left } t'} \quad (\text{E-LEFT})$	$\frac{t \rightarrow t'}{\text{right } t \rightarrow \text{right } t'} \quad (\text{E-RIGHT})$
$\frac{t \rightarrow t'}{\text{match } t \text{ with left } x_1 \rightarrow t_1 \mid \text{right } x_2 \rightarrow t_2 \rightarrow \text{match } t' \text{ with left } x_1 \rightarrow t_1 \mid \text{right } x_2 \rightarrow t_2} \quad (\text{E-MATCH})$	
$\frac{}{\text{match left } v \text{ with left } x_1 \rightarrow t_1 \mid \text{right } x_2 \rightarrow t_2 \rightarrow [x_1 \mapsto v]t_1} \quad (\text{E-MATCHLEFT})$	
$\frac{}{\text{match right } v \text{ with left } x_1 \rightarrow t_1 \mid \text{right } x_2 \rightarrow t_2 \rightarrow [x_2 \mapsto v]t_2} \quad (\text{E-MATCHRIGHT})$	
$\frac{t_1 \rightarrow t'_1}{\text{let } x=t_1 \text{ in } t_2 \rightarrow \text{let } x=t'_1 \text{ in } t_2} \quad (\text{E-LET})$	$\frac{}{\text{let } x=v \text{ in } t \rightarrow [x \mapsto v]t} \quad (\text{E-LETRED})$
$\frac{}{\text{letrec } x=v \text{ in } t \rightarrow \text{let } x=[x \mapsto \text{letrec } x=v \text{ in } x]v \text{ in } t} \quad (\text{E-LETREC})$	

3.3 Static Type System

$\frac{}{\Gamma \vdash () : \text{True}}$	(T-TRUE)
$\frac{\Gamma \vdash t : \text{False}}{\Gamma \vdash t : T}$	(T-FALSE)
$\frac{\Gamma \vdash t_1 : T_1 \quad \Gamma \vdash t_2 : T_2}{\Gamma \vdash (t_1, t_2) : T_1 \wedge T_2}$	(T-PAIR)

$$\begin{array}{c}
\frac{\Gamma \vdash t : T_1 \wedge T_2}{\Gamma \vdash \text{fst } t : T_1} \quad (\text{T-FST}) \qquad \frac{\Gamma \vdash t : T_1 \wedge T_2}{\Gamma \vdash \text{snd } t : T_2} \quad (\text{T-SND}) \\
\\
\frac{\Gamma \vdash t : T_1}{\Gamma \vdash \text{left } t : T_1 \vee T_2} \quad (\text{T-LEFT}) \qquad \frac{\Gamma \vdash t : T_2}{\Gamma \vdash \text{right } t : T_1 \vee T_2} \quad (\text{T-RIGHT}) \\
\\
\frac{\Gamma \vdash t : T_1 \vee T_2 \quad \Gamma, x_1:T_1 \vdash t_1 : T \quad \Gamma, x_2:T_2 \vdash t_2 : T}{\Gamma \vdash \text{match } t \text{ with left } x_1 \rightarrow t_1 \mid \text{right } x_2 \rightarrow t_2 : T} \quad (\text{T-MATCH}) \\
\\
\frac{\Gamma \vdash t_1 : T_1 \quad \Gamma, x:T_1 \vdash t_2 : T}{\Gamma \vdash \text{let } x=t_1 \text{ in } t_2 : T} \quad (\text{T-LET}) \qquad \frac{\Gamma, x:T_1 \vdash v_1 : T_1 \quad \Gamma, x:T_1 \vdash t_2 : T}{\Gamma \vdash \text{letrec } x=v_1 \text{ in } t_2 : T} \quad (\text{T-LETREC})
\end{array}$$

4 Mutable References

4.1 Syntax

$t ::= \text{ref } t \mid !t \mid t := t \mid l$
 $v ::= l \mid ()$
 $l ::= \text{memory locations}$
 $T ::= \text{Ref } T$

4.2 Small-Step Operational Semantics

μ is a finite function from memory locations to values.

$$\begin{array}{c}
\frac{\langle t_1, \mu \rangle \longrightarrow \langle t'_1, \mu' \rangle}{\langle \text{ref } t_1, \mu \rangle \longrightarrow \langle \text{ref } t'_1, \mu' \rangle} \quad (\text{E-REF}) \qquad \frac{l \notin \text{domain}(\mu)}{\langle \text{ref } v, \mu \rangle \longrightarrow \langle l, \mu \cup \{(l, v)\} \rangle} \quad (\text{E-REFV}) \\
\\
\frac{\langle t_1, \mu \rangle \longrightarrow \langle t'_1, \mu' \rangle}{\langle !t_1, \mu \rangle \longrightarrow \langle !t'_1, \mu' \rangle} \quad (\text{E-DEREF}) \qquad \frac{\mu(l)=v}{\langle !l, \mu \rangle \longrightarrow \langle v, \mu \rangle} \quad (\text{E-DEREFLOC}) \\
\\
\frac{\langle t_1, \mu \rangle \longrightarrow \langle t'_1, \mu' \rangle}{\langle t_1 := t_2, \mu \rangle \longrightarrow \langle t'_1 := t_2, \mu' \rangle} \quad (\text{E-ASSIGN1}) \qquad \frac{\langle t_2, \mu \rangle \longrightarrow \langle t'_2, \mu' \rangle}{\langle v_1 := t_2, \mu \rangle \longrightarrow \langle v_1 := t'_2, \mu' \rangle} \quad (\text{E-ASSIGN2}) \\
\\
\frac{l \in \text{domain}(\mu)}{\langle l := v, \mu \rangle \longrightarrow \langle (), [l \mapsto v] \mu \rangle} \quad (\text{E-ASSIGN})
\end{array}$$

4.3 Type System

Σ is a finite function from memory locations to types.

$$\begin{array}{c}
\frac{\Sigma(l) = T}{\Gamma; \Sigma \vdash l : \text{Ref } T} \quad (\text{T-LOC}) \\
\\
\frac{\Gamma; \Sigma \vdash t_1 : T_1}{\Gamma; \Sigma \vdash \text{ref } t_1 : \text{Ref } T_1} \quad (\text{T-REF}) \\
\\
\frac{\Gamma; \Sigma \vdash t : \text{Ref } T}{\Gamma; \Sigma \vdash !t : T} \quad (\text{T-DEREF}) \\
\\
\frac{\Gamma; \Sigma \vdash t_1 : \text{Ref } T \quad \Gamma; \Sigma \vdash t_2 : T}{\Gamma; \Sigma \vdash t_1 := t_2 : \text{True}} \quad (\text{T-ASSIGN})
\end{array}$$