

$t ::=$	terms	$T ::=$	types
true	constant true	Int	type of integers
false	constant false	Bool	type of booleans
n	numeric value	$T \rightarrow T$	type of functions
x	variable		
$\lambda x : T. t$	abstraction	$T^* ::=$	multi – types
$t_1 t_2$	application	$\{\overline{T}\}$	multi – type
olet $x : T = t$ in t	overloading let	$\Gamma ::=$	typing contexts
$t :: T$	ascription	\emptyset	empty context
		$\Gamma, x : T$	term variable binding
$v ::=$	values	$\phi ::=$	multi – typing contexts
true	true value	\emptyset	empty context
false	false value	$\phi, x : T^*$	term variable binding
n	numeric value		
$(\lambda x : T. t)[s]$	closure	$s ::=$	explicit substitutions
		\bullet	empty substitution
$c ::=$	configurations	$x \mapsto \{(\overline{v : T})\}, s$	variable substitution
v			
$t[s]$			
$c :: T$			
olet $x : T = c$ in c			
$c c$			

Figure 1: Syntax of the simply typed lambda-calculus with overloading.

Lemma 1 (Inversion of term typing).

1. If $\Gamma; \phi \vdash x : R$, then $x : R \in \Gamma$
2. If $\Gamma; \phi \vdash \lambda x : T_1. t_2 : R$, then $R = T_1 \rightarrow R_2$ for some R_2 , with $\Gamma, x : T_1; \phi \vdash t_2 : R_2$.
3. If $\Gamma; \phi \vdash t_1 t_2 : R$, then there is some type T_{11} such that $\Gamma; \phi \vdash t_1 : T_{11} \rightarrow R$ and $\Gamma; \phi \vdash t_2 : T_{11}$.

Proof. Immediate from the definition of the typing relation. \square

Lemma 2 (Inversion of configuration typing).

1. If $\Gamma \vdash_c x[s] : R$, then $(x, v) \in s$, for some v , and $\Gamma \vdash_c v : R$.
2. If $\Gamma \vdash_c (\lambda x : T_1. t_2)[s] : R$, then $R = T_1 \rightarrow R_2$ for some R_2 , with $\Gamma, x : T_1; \phi(s) \vdash t_2 : R_2$.
3. If $\Gamma \vdash_c (t_1 t_2)[s] : R$, then $\Gamma \vdash_c t_1[s] t_2[s] : R$.
4. If $\Gamma \vdash_c c_1 c_2 : R$, then there is some type T_{11} such that $\Gamma \vdash_c c_1 : T_{11} \rightarrow R$ and $\Gamma \vdash_c c_2 : T_{11}$.

$\Gamma; \phi \vdash t \xrightarrow{\rightarrow} T^* \Rightarrow t$	$\Gamma; \phi \vdash t \xleftarrow{\leftarrow} T \Rightarrow t$
$\Gamma; \phi \vdash \text{true} \xrightarrow{\rightarrow} \{\text{Bool}\}$ (STTrue)	$\Gamma; \phi \vdash \text{true} \xleftarrow{\leftarrow} \text{Bool}$ (CTTrue)
$\Gamma; \phi \vdash \text{false} \xrightarrow{\rightarrow} \{\text{Bool}\}$ (STFalse)	$\Gamma; \phi \vdash \text{false} \xleftarrow{\leftarrow} \text{Bool}$ (CTFalse)
$\Gamma; \phi \vdash n \xrightarrow{\rightarrow} \{\text{Int}\}$ (STNum)	$\Gamma; \phi \vdash n \xleftarrow{\leftarrow} \text{Int}$ (CTNum)
$\frac{x : T \in \Gamma}{\Gamma; \phi \vdash x \xrightarrow{\rightarrow} \{T\}}$ (STVar Γ)	$\frac{\Gamma; \phi \vdash x \xrightarrow{\rightarrow} \{T\}}{\Gamma; \phi \vdash x \xleftarrow{\leftarrow} T}$ (CTVar Γ)
$\frac{x : T^* \in \phi}{\Gamma; \phi \vdash x \xrightarrow{\rightarrow} T^*}$ (STVar ϕ)	$\frac{\Gamma; \phi \vdash x \xrightarrow{\rightarrow} T^* \quad T \in T^*}{\Gamma; \phi \vdash x \xleftarrow{\leftarrow} T \Rightarrow x :: T}$ (CTVar ϕ)
$\frac{\Gamma; \phi \vdash t \xleftarrow{\leftarrow} T}{\Gamma; \phi \vdash t :: T \xrightarrow{\rightarrow} \{T\}}$ (STAsc)	$\frac{\Gamma; \phi \vdash t \xleftarrow{\leftarrow} T}{\Gamma; \phi \vdash t :: T \xleftarrow{\leftarrow} T}$ (CTAsc)
$\frac{x \notin \text{dom}(\Gamma) \quad \Gamma; \phi \vdash t_1 \xrightarrow{\rightarrow} T_1^* \quad T_1 \in T_1^* \quad \Gamma; \phi \oplus (x : T_1) \vdash t_2 \xrightarrow{\rightarrow} T_2^*}{\Gamma; \phi \vdash \text{olet } x : T_1 = t_1 \text{ in } t_2 \xrightarrow{\rightarrow} T_2^*}$ (STOLet)	$\frac{x \notin \text{dom}(\Gamma) \quad \Gamma; \phi \vdash t_1 \xrightarrow{\rightarrow} T_1^* \quad T_1 \in T_1^* \quad \Gamma; \phi \oplus (x : T_1) \vdash t_2 \xleftarrow{\leftarrow} T_2}{\Gamma; \phi \vdash \text{olet } x : T_1 = t_1 \text{ in } t_2 \xleftarrow{\leftarrow} T_2}$ (CTOLet)
$\frac{x \notin \text{dom}(\Gamma \cup \phi) \quad \Gamma, x : T_1; \phi \vdash t_2 \xrightarrow{\rightarrow} T_2^*}{\Gamma; \phi \vdash \lambda x : T_1. t_2 \xrightarrow{\rightarrow} \ T_1 \rightarrow T_2\ }$ (STAbs)	$\frac{x \notin \text{dom}(\Gamma \cup \phi) \quad \Gamma, x : T_1; \phi \vdash t_2 \xleftarrow{\leftarrow} T_2}{\Gamma; \phi \vdash \lambda x : T_1. t_2 \xleftarrow{\leftarrow} T_1 \rightarrow T_2}$ (CTAbs)
$\frac{\Gamma; \phi \vdash t_1 \xrightarrow{\rightarrow} T^* \quad \exists! T_1 \rightarrow T_2 \in T^* \mid \Gamma; \phi \vdash t_2 \xleftarrow{\leftarrow} T_1}{\Gamma; \phi \vdash t_1 t_2 \xrightarrow{\rightarrow} \{T_2\} \Rightarrow (t_1 :: T_1 \rightarrow T_2) (t_2 :: T_1)}$ (STApp)	$\frac{\Gamma; \phi \vdash t_1 \xrightarrow{\rightarrow} T^* \quad \exists! T_1 \rightarrow T_2 \in T^* \mid \Gamma; \phi \vdash t_2 \xleftarrow{\leftarrow} T_1}{\Gamma; \phi \vdash t_1 t_2 \xleftarrow{\leftarrow} T_2 \Rightarrow (t_1 :: T_1 \rightarrow T_2) (t_2 :: T_1)}$ (CTApp)

Figure 2: Type synthesis and checking.

$\Gamma \vdash_c c \xrightarrow{\rightarrow} T^* \Rightarrow c$	$\Gamma \vdash_c c \xleftarrow{\leftarrow} T \Rightarrow c$
$\Gamma \vdash_c \text{true} \xrightarrow{\rightarrow} \{\text{Bool}\}$ (STCTrue)	$\Gamma \vdash_c \text{true} \xleftarrow{\leftarrow} \text{Bool}$ (CTCTrue)
$\Gamma \vdash_c \text{false} \xrightarrow{\rightarrow} \{\text{Bool}\}$ (STCFalse)	$\Gamma \vdash_c \text{false} \xleftarrow{\leftarrow} \text{Bool}$ (CTCFalse)
$\Gamma \vdash_c \text{true}[s] \xrightarrow{\rightarrow} \{\text{Bool}\}$ (STCCTrue)	$\Gamma \vdash_c \text{true}[s] \xleftarrow{\leftarrow} \text{Bool}$ (CTCCTrue)
$\Gamma \vdash_c \text{false}[s] \xrightarrow{\rightarrow} \{\text{Bool}\}$ (STCCFalse)	$\Gamma \vdash_c \text{false}[s] \xleftarrow{\leftarrow} \text{Bool}$ (CTCCFalse)
$\Gamma \vdash_c n \xrightarrow{\rightarrow} \{\text{Int}\}$ (STCNum)	$\Gamma \vdash_c n \xleftarrow{\leftarrow} \text{Int}$ (CTCNum)
$\Gamma \vdash_c n[s] \xrightarrow{\rightarrow} \{\text{Int}\}$ (STCCNum)	$\Gamma \vdash_c n[s] \xleftarrow{\leftarrow} \text{Int}$ (CTCCNum)
$\frac{x : T \in \Gamma}{\Gamma \vdash_c x[s] \xrightarrow{\rightarrow} \{T\}}$ (STCVar Γ)	$\frac{\Gamma \vdash_c x[s] \xrightarrow{\rightarrow} \{T\}}{\Gamma \vdash_c x[s] \xleftarrow{\leftarrow} T}$ (CTCVar Γ)
$\frac{x, \{\overline{(v : T)}\} \in s}{\Gamma \vdash_c x[s] \xrightarrow{\rightarrow} \{\overline{T}\}}$ (STCVar ϕ)	$\frac{x, \{\overline{(v : T)}\} \in s \quad T \in \{\overline{T}\}}{\Gamma \vdash_c x[s] \xleftarrow{\leftarrow} T \Rightarrow (x :: T)[s]}$ (CTCVar ϕ)
$\frac{\Gamma \vdash_c t[s] :: T \xrightarrow{\rightarrow} T^*}{\Gamma \vdash_c (t :: T)[s] \xrightarrow{\rightarrow} T^*}$ (STCAsc)	$\frac{\Gamma \vdash_c t[s] :: T \xleftarrow{\leftarrow} T}{\Gamma \vdash_c (t :: T)[s] \xleftarrow{\leftarrow} T}$ (CTCAsc)
$\frac{\Gamma \vdash_c c \xleftarrow{\leftarrow} T}{\Gamma \vdash_c c :: T \xrightarrow{\rightarrow} \{T\}}$ (STCCAsc)	$\frac{\Gamma \vdash_c c \xleftarrow{\leftarrow} T}{\Gamma \vdash_c c :: T \xleftarrow{\leftarrow} \{T\}}$ (CTCCAsc)
$\frac{\Gamma \vdash_c \text{olet } x : T_1 = t_1[s] \text{ in } t_2[s] \xrightarrow{\rightarrow} T^*}{\Gamma \vdash_c (\text{olet } x : T_1 = t_1 \text{ in } t_2)[s] \xrightarrow{\rightarrow} T^*}$ (STCOlet)	$\frac{\Gamma \vdash_c \text{olet } x : T_1 = t_1[s] \text{ in } t_2[s] \xleftarrow{\leftarrow} T}{\Gamma \vdash_c (\text{olet } x : T_1 = t_1 \text{ in } t_2)[s] \xleftarrow{\leftarrow} T^*}$ (CTCOlet)
$\frac{x \notin \text{dom}(\Gamma) \quad \Gamma \vdash_c c_1 \xrightarrow{\rightarrow} T_1^* \quad T_1 \in T_1^* \quad \Gamma; \phi(s) \oplus (x : T_1) \vdash t_2 \xrightarrow{\rightarrow} T_2^*}{\Gamma \vdash_c \text{olet } x : T_1 = c_1 \text{ in } t_2[s] \xrightarrow{\rightarrow} T_2^*}$ (STCCOlet)	$\frac{x \notin \text{dom}(\Gamma) \quad \Gamma \vdash_c c_1 \xrightarrow{\rightarrow} T_1^* \quad T_1 \in T_1^* \quad \Gamma; \phi(s) \oplus (x : T_1) \vdash t_2 \xleftarrow{\leftarrow} T_2}{\Gamma \vdash_c \text{olet } x : T_1 = c_1 \text{ in } t_2[s] \xleftarrow{\leftarrow} T_2}$ (CTCCOlet)
$\frac{x \notin \text{dom}(\Gamma \cup \phi(s)) \quad \Gamma, x : T_1; \phi(s) \vdash t_2 \xrightarrow{\rightarrow} T_2^*}{\Gamma \vdash_c (\lambda x : T_1. t_2)[s] \xrightarrow{\rightarrow} \ T_1 \rightarrow T_2^*\ }$ (STCAbs)	$\frac{x \notin \text{dom}(\Gamma \cup \phi(s)) \quad \Gamma, x : T_1; \phi(s) \vdash t_2 \xleftarrow{\leftarrow} T_2}{\Gamma \vdash_c (\lambda x : T_1. t_2)[s] \xleftarrow{\leftarrow} T_1 \rightarrow T_2}$ (CTCAbs)
$\frac{\Gamma \vdash_c t_1[s] \quad t_2[s] \xrightarrow{\rightarrow} T^*}{\Gamma \vdash_c (t_1 \ t_2)[s] \xrightarrow{\rightarrow} T^*}$ (STCApp)	$\frac{\Gamma \vdash_c t_1[s] \quad t_2[s] \xleftarrow{\leftarrow} T}{\Gamma \vdash_c (t_1 \ t_2)[s] \xleftarrow{\leftarrow} T}$ (CTCApp)
$\frac{\Gamma \vdash_c c_1 \xrightarrow{\rightarrow} T^* \quad \exists! T_1 \rightarrow T_2 \in T^* \mid \Gamma \vdash_c c_2 \xleftarrow{\leftarrow} T_1}{\Gamma \vdash_c c_1 \ c_2 \xrightarrow{\rightarrow} \{T_2\} \Rightarrow (c_1 :: T_1 \rightarrow T_2) \ (c_2 :: T_1)}$ (STCCApp) ₃	$\frac{\Gamma \vdash_c c_1 \xrightarrow{\rightarrow} T^* \quad \exists! T_1 \rightarrow T_2 \in T^* \mid \Gamma \vdash_c c_2 \xleftarrow{\leftarrow} T_1}{\Gamma \vdash_c c_1 \ c_2 \xleftarrow{\leftarrow} T_2 \Rightarrow (c_1 :: T_1 \rightarrow T_2) \ (c_2 :: T_1)}$ (CTCCApp)

Figure 3: Configuration synthesis and checking.

	$c \longrightarrow c$	
$\text{true}[s] \longrightarrow \text{true}$		(True)
$\text{false}[s] \longrightarrow \text{false}$		(False)
$n[s] \longrightarrow n$		(Num)
$x[x, \{(v : T_1)\} : s] \longrightarrow v$		(VarOk)
$\frac{x \neq y}{x[y \mapsto \{(v : T_1)\}, s] \longrightarrow x[s]}$		(VarNext)
$\frac{(v_i, T_i) \in \{(\overline{v : T})\}}{x[x \mapsto \{(v : T)\}, s] :: T_i \longrightarrow v_i}$		(VarAscOk)
$\frac{x \neq y}{x[y \mapsto \{(\overline{v : T_1})\}, s] :: T \longrightarrow x[s] :: T}$		(VarAscNext)
$(t :: T)[s] \longrightarrow t[s] :: T$		(AscSub)
$v :: T \longrightarrow v$		(Asc)
$\frac{c \longrightarrow c'}{c :: T \longrightarrow c' :: T}$		(Asc1)
$(\text{olet } x : T_1 = t_1 \text{ in } t_2)[s] \longrightarrow \text{olet } x : T_1 = t_1[s] \text{ in } t_2[s]$		(LetSub)
$\text{olet } x : T_1 = v \text{ in } t_2[s] \longrightarrow t_2[x, (v : T_1) :^* s]$		(Let)
$\frac{c_1 \longrightarrow c'_1}{\text{olet } x : T_1 = c_1 \text{ in } t_2[s] \longrightarrow \text{olet } x : T_1 = c'_1 \text{ in } t_2[s]}$		(Let1)
$(t_1 \ t_2)[s] \longrightarrow t_1[s] \ t_2[s]$		(AppSub)
$(\lambda x : T_1. t_2)[s] \ v \longrightarrow ([x \mapsto v]t_2)[s]$		(App)
$\frac{c_1 \longrightarrow c'_1}{c_1 \ c_2 \longrightarrow c'_1 \ c_2}$		(App1)
$\frac{c \longrightarrow c'}{v \ c \longrightarrow v \ c'}$		(App2)

Figure 4: Configuration reduction rules.

Proof. Immediate from the definition of the typing relation. \square

Lemma 3 (Canonical Forms).

1. If v is a value of type $T_1 \rightarrow T_2$, then $v = (\lambda x : T_1. t_2)[s]$.

Proof. Straightforward. \square

Definition 1 ($\Gamma(s)$). The typing context built from a substitution s , writing $\Gamma(s)$, it is defined as follows:

$$\Gamma(s) = \begin{cases} \emptyset & s = \bullet \\ \Gamma(s'), x : T & s = (x, v) : s' \wedge \Gamma \vdash_c v : T \end{cases}$$

Definition 2 (\oplus). Given a multi-type context ϕ and a pair $(x : T)$, the operator \oplus is defined as follows:

$$\phi \oplus (x : T) = \begin{cases} x : \{T\} & \phi = \emptyset \\ \phi', x : (T^* \cup \{T\}) & \phi = \phi', x : T^* \\ \phi' \oplus (x : T), y : T^* & \phi = \phi', y : T^* \end{cases}$$

Definition 3 ($\|\cdot\|$). Given $T_1 \rightarrow T_2^*$, the operator $\|\cdot\|$ is defined as follows:

$$\|T_1 \rightarrow T_2^*\| = \{T_1 \rightarrow T_2 \mid T_2 \in T_2^*\}$$

Theorem 4 (Progress). Suppose c is a well-typed configuration (that is, $\Gamma \vdash_c c : T$ for some T). Then either c is a value or else there is some c' such that $c \longrightarrow c'$.

Proof. By induction on a derivation of $\Gamma \vdash_c c : T$.

Case (TCVar). Then $c = x[s]$, with $(x, v) \in s$, for some v , and $\Gamma \vdash_c v : T$. Since $x \in \text{dom}(s)$, if the substitution $s = x \mapsto \{(v : T)\}, s'$, then rule *VarOk*, applies, otherwise, rule *VarNext* applies.

Case (TCAbs). Then $c = (\lambda x : T_1. t_2)[s]$. This case is immediate, since closures are values.

Case (TCApp). Then $c = (t_1 \ t_2)[s]$, so rule *AppSub* applies to c .

Case (TCCApp). Then $c = c_1 \ c_2$, with $\Gamma \vdash_c c_1 : T_{11} \rightarrow T$, for some T_{11} and $\Gamma \vdash_c c_2 : T_{11}$, by the Lemma 2. Then, by the induction hypothesis, either c_1 is a value or else it can take a step of evaluation, and likewise c_2 . If c_1 can take a step, then rule *App1* applies to c . If c_1 is a value and c_2 can take a step, then rule *App2* applies. Finally, if both c_1 and c_2 are values, then the Lemma 3 tells us that c_1 has the form $(\lambda x : T_{11}. t_{12})[s]$, and so rule *App* applies to c . \square

Definition 4 (Well typed substitution). A substitution s is said well typed with a typing context Γ , writing $\Gamma; \phi \vdash s$, if $\text{dom}(s) = \text{dom}(\Gamma)$ and for every $(x, v) \in s$ and $\Gamma \vdash_c v : T$, where $x : T \in \Gamma$.

Lemma 5 (Permutation). *If $\Gamma; \phi \vdash t : T$ and Δ is a permutation of Γ , then $\Delta \vdash t : T$.*

Proof. By induction on typing derivations. \square

Lemma 6. *If $\Gamma; \phi \vdash s$ then Γ is a permutation of $\Gamma(s)$.*

Proof. By the definition of well typed substitution. \square

Lemma 7. *If $\Gamma; \phi \vdash s$ and $\Gamma \vdash_c v : T$, then $\Gamma, x : T \vdash x \mapsto \{\overline{(v : T_1)}\}, s$.*

Proof. By the definition of well typed substitution. \square

Lemma 8. *If $\Gamma; \phi \vdash s$ then $\Gamma \vdash_c t[s] : T$ if and only if $\Gamma; \phi \vdash t : T$.*

Proof. By induction on typing derivations, using Lemma 5 and Lemma 6. \square

Theorem 9 (Preservation). *If $\Gamma \vdash_c c : T$ and $c \longrightarrow c'$, then $\Gamma \vdash_c c' : T$.*

Proof. By induction on a derivation of $\Gamma \vdash_c c : T$.

Case (TCVar). Then $c = x[s]$, with $\Gamma \vdash_c (x, v) \in s$, for some v , and $\Gamma \vdash_c v : T$. We find that there are two rule by which $c \longrightarrow c'$ can be derived: *VarOk* and *VarNext*. We consider each case separately.

- *Subcase (VarOk).* Then $s = x \mapsto \{\overline{(v : T)}\}, s'$ and $c' = v$. Since $(x, v) \in s$ and $\Gamma \vdash_c v : T$, then $\Gamma \vdash_c c' : T$.
- *Subcase (VarNext).* Then $s = y \mapsto \{\overline{(v : T)}\}, s'$, $x \neq y$ and $c' = x[s']$. Since $(x, v) \in s'$ too, and $\Gamma \vdash_c v : T$ then $\Gamma \vdash_c x[s'] : T$, that is $\Gamma \vdash_c c' : T$.

Case (TAbs). Then $c = (\lambda x : T_1. t_2)[s]$. It cannot be the case that $c \longrightarrow c'$, because c is a value, then the requirements of the theorem are vacuously satisfied.

Case (TCApp). Then $c = (t_1 t_2)[s]$ and $\Gamma \vdash_c t_1[s] t_2[s] : T$. We find that there are only one rule by which $c \longrightarrow c'$ can be derived: *AppSub*. With this rule $c' = t_1[s] t_2[s]$, then we can conclude that $\Gamma \vdash_c c' : T$.

Case (TCCApp). Then $c = c_1 c_2$, $\Gamma \vdash_c c_1 : T_2 \rightarrow T$ and $\Gamma \vdash_c c_2 : T_2$. We find that there are three rules by which $c \longrightarrow c'$ can be derived: *App1*, *App2* and *App*. We consider each case separately.

- *Subcase (App1).* Then $c_1 \longrightarrow c'_1$, $c' = c'_1 c_2$. By the induction hypothesis, $\Gamma \vdash_c c'_1 : T_2 \rightarrow T$, then we can apply rule *TCCApp*, to conclude that $\Gamma \vdash_c c'_1 c_2 : T$, that is $\Gamma \vdash_c c' : T$.
- *Subcase (App2).* Then $c_2 \longrightarrow c'_2$, $c' = c_1 c'_2$. By the induction hypothesis, $\Gamma \vdash_c c'_2 : T_2$, then we can apply rule *TCCApp*, to conclude that $\Gamma \vdash_c c_1 c'_2 : T$, that is $\Gamma \vdash_c c' : T$.
- *Subcase (App):* Then $c_1 = (\lambda x : T_1. t_{12})[s]$, $c_2 = v$, $c' = t_{12}[x \mapsto \{\overline{(v : T_1)}\}, s]$ and $\Gamma, x : T_1; \phi(s) \vdash t_{12} : T$ by the Lemma 2. Since we know that $\Gamma, x : T_1; \phi(s) \vdash x \mapsto \{\overline{(v : T_1)}\}, s$ by the Lemma 7, the resulting configuration $\Gamma \vdash_c t_{12}[x \mapsto \{\overline{(v : T_1)}\}, s] : T$, by the Lemma 8, that is $\Gamma \vdash_c c' : T$.

\square