t	::=		terms	T	::=		types
		true	constant true			Int	type of integers
		false	constant false			Bool	type of booleans
		n	numeric value			$T \to T$	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				
		$t :: T$	ascription	Γ	::=		typing contexts
						$\varnothing$	empty context
v	::=		values			$\Gamma, x : T$	term variable binding
		true	true value				
		false	false value	ϕ	::=		multi − typing contexts
		n	numeric value			$\varnothing$	empty context
		$(\lambda x : T.\ t)[s]$	closure			$\phi, x : T^*$	term variable binding
c	::=		configurations	s	::=		explicit substitutions
		v				$\bullet$	empty substitution
		$t[s]$				$x \mapsto \{(\overline{v : T})\}, s$	variable substitution
		$c :: T$					
		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* $R = T_1 \to R_2$ *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* $\Gamma; \phi \vdash t_1 : T_{11} \to R$ *and* $\Gamma; \phi \vdash t_2 : T_{11}$*.*

Proof. Immediate from the definition of the typing relation. □

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* $R = T_1 \to R_2$ *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* $\Gamma \vdash_c c_1 : T_{11} \to R$ *and* $\Gamma \vdash_c c_2 : T_{11}$*.*

$$\boxed{\Gamma; \phi \vdash t \overrightarrow{\in} T^* \Rightarrow t}$$

$$\Gamma; \phi \vdash \mathsf{true} \overrightarrow{\in} \{\mathsf{Bool}\} \quad \text{(STTrue)}$$

$$\Gamma; \phi \vdash \mathsf{false} \overrightarrow{\in} \{\mathsf{Bool}\} \quad \text{(STFalse)}$$

$$\Gamma; \phi \vdash n \overrightarrow{\in} \{\mathsf{Int}\} \quad \text{(STNum)}$$

$$\frac{x : T \in \Gamma}{\Gamma; \phi \vdash x \overrightarrow{\in} \{T\}} \quad \text{(STVar}\Gamma\text{)}$$

$$\frac{x : T^* \in \phi}{\Gamma; \phi \vdash x \overrightarrow{\in} T^*} \quad \text{(STVar}\phi\text{)}$$

$$\frac{\Gamma; \phi \vdash t \overleftarrow{\in} T}{\Gamma; \phi \vdash t :: T \overrightarrow{\in} \{T\}} \quad \text{(STAsc)}$$

$$\frac{\begin{array}{c} x \notin \mathit{dom}(\Gamma) \qquad \Gamma; \phi \vdash t_1 \overrightarrow{\in} T_1^* \\ T_1 \in T_1^* \qquad \Gamma; \phi \oplus (x : T_1) \vdash t_2 \overrightarrow{\in} T_2^* \end{array}}{\Gamma; \phi \vdash \mathsf{olet}\ x : T_1 = t_1\ \mathsf{in}\ t_2 \overrightarrow{\in} T_2^*} \quad \text{(STOLet)}$$

$$\frac{x \notin \mathit{dom}(\Gamma \cup \phi) \qquad \Gamma, x : T_1; \phi \vdash t_2 \overrightarrow{\in} T_2^*}{\Gamma; \phi \vdash \lambda x : T_1.\ t_2 \overrightarrow{\in} \|T_1 \to T_2\|} \quad \text{(STAbs)}$$

$$\frac{\begin{array}{c} \Gamma; \phi \vdash t_1 \overrightarrow{\in} T^* \\ \exists!\ T_1 \to T_2 \in T^* \mid \Gamma; \phi \vdash t_2 \overleftarrow{\in} T_1 \end{array}}{\Gamma; \phi \vdash t_1\ t_2 \overrightarrow{\in} \{T_2\} \Rightarrow (t_1 :: T_1 \to T_2)\ (t_2 :: T_1)} \quad \text{(STApp)}$$

$$\boxed{\Gamma; \phi \vdash t \overleftarrow{\in} T \Rightarrow t}$$

$$\Gamma; \phi \vdash \mathsf{true} \overleftarrow{\in} \mathsf{Bool} \quad \text{(CTTrue)}$$

$$\Gamma; \phi \vdash \mathsf{false} \overleftarrow{\in} \mathsf{Bool} \quad \text{(CTFalse)}$$

$$\Gamma; \phi \vdash n \overleftarrow{\in} \mathsf{Int} \quad \text{(CTNum)}$$

$$\frac{\Gamma; \phi \vdash x \overrightarrow{\in} \{T\}}{\Gamma; \phi \vdash x \overleftarrow{\in} T} \quad \text{(CTVar}\Gamma\text{)}$$

$$\frac{\Gamma; \phi \vdash x \overrightarrow{\in} T^* \qquad T \in T^*}{\Gamma; \phi \vdash x \overleftarrow{\in} T \Rightarrow x :: T} \quad \text{(CTVar}\phi\text{)}$$

$$\frac{\Gamma; \phi \vdash t \overleftarrow{\in} T}{\Gamma; \phi \vdash t :: T \overleftarrow{\in} T} \quad \text{(CTAsc)}$$

$$\frac{\begin{array}{c} x \notin \mathit{dom}(\Gamma) \qquad \Gamma; \phi \vdash t_1 \overrightarrow{\in} T_1^* \\ T_1 \in T_1^* \qquad \Gamma; \phi \oplus (x : T_1) \vdash t_2 \overleftarrow{\in} T_2 \end{array}}{\Gamma; \phi \vdash \mathsf{olet}\ x : T_1 = t_1\ \mathsf{in}\ t_2 \overleftarrow{\in} T_2} \quad \text{(CTOLet)}$$

$$\frac{x \notin \mathit{dom}(\Gamma \cup \phi) \qquad \Gamma, x : T_1; \phi \vdash t_2 \overleftarrow{\in} T_2}{\Gamma; \phi \vdash \lambda x : T_1.\ t_2 \overleftarrow{\in} T_1 \to T_2} \quad \text{(CTAbs)}$$

$$\frac{\begin{array}{c} \Gamma; \phi \vdash t_1 \overrightarrow{\in} T^* \\ \exists!\ T_1 \to T_2 \in T^* \mid \Gamma; \phi \vdash t_2 \overleftarrow{\in} T_1 \end{array}}{\Gamma; \phi \vdash t_1\ t_2 \overleftarrow{\in} T_2 \Rightarrow (t_1 :: T_1 \to T_2)\ (t_2 :: T_1)} \quad \text{(CTApp)}$$

Figure 2: Type synthesis and checking.

$$\boxed{\Gamma \vdash_c c \overrightarrow{\in} T^* \Rightarrow c}$$

$$\Gamma \vdash_c \mathsf{true} \overrightarrow{\in} \{\mathsf{Bool}\} \qquad \text{(STCTrue)}$$

$$\Gamma \vdash_c \mathsf{false} \overrightarrow{\in} \{\mathsf{Bool}\} \qquad \text{(STCFalse)}$$

$$\Gamma \vdash_c \mathsf{true}[s] \overrightarrow{\in} \{\mathsf{Bool}\} \qquad \text{(STCCTrue)}$$

$$\Gamma \vdash_c \mathsf{false}[s] \overrightarrow{\in} \{\mathsf{Bool}\} \qquad \text{(STCCFalse)}$$

$$\Gamma \vdash_c n \overrightarrow{\in} \{\mathsf{Int}\} \qquad \text{(STCNum)}$$

$$\Gamma \vdash_c n[s] \overrightarrow{\in} \{\mathsf{Int}\} \qquad \text{(STCCNum)}$$

$$\frac{x : T \in \Gamma}{\Gamma \vdash_c x[s] \overrightarrow{\in} \{T\}} \qquad \text{(STCVar}\Gamma\text{)}$$

$$\frac{x, \{(\overline{v : T})\} \in s}{\Gamma \vdash_c x[s] \overrightarrow{\in} \{\overline{T}\}} \qquad \text{(STCVar}\phi\text{)}$$

$$\frac{\Gamma \vdash_c t[s] :: T \overrightarrow{\in} T^*}{\Gamma \vdash_c (t :: T)[s] \overrightarrow{\in} T^*} \qquad \text{(STCAsc)}$$

$$\frac{\Gamma \vdash_c c \overleftarrow{\in} T}{\Gamma \vdash_c c :: T \overrightarrow{\in} \{T\}} \qquad \text{(STCCAsc)}$$

$$\frac{\Gamma \vdash_c \mathsf{olet}\ x : T_1 = t_1[s]\ \mathsf{in}\ t_2[s] \overrightarrow{\in} T^*}{\Gamma \vdash_c (\mathsf{olet}\ x : T_1 = t_1\ \mathsf{in}\ t_2)[s] \overrightarrow{\in} T^*} \qquad \text{(STCOLet)}$$

$$\frac{\begin{array}{c} x \notin dom(\Gamma) \qquad \Gamma \vdash_c c_1 \overrightarrow{\in} T_1^* \\ T_1 \in T_1^* \qquad \Gamma; \phi(s) \oplus (x : T_1) \vdash t_2 \overrightarrow{\in} T_2^* \end{array}}{\Gamma \vdash_c \mathsf{olet}\ x : T_1 = c_1\ \mathsf{in}\ t_2[s] \overrightarrow{\in} T_2^*} \qquad \text{(STCCOLet)}$$

$$\frac{x \notin dom(\Gamma \cup \phi(s)) \qquad \Gamma, x : T_1; \phi(s) \vdash t_2 \overrightarrow{\in} T_2^*}{\Gamma \vdash_c (\lambda x : T_1.\ t_2)[s] \overrightarrow{\in} \|T_1 \to T_2^*\|} \qquad \text{(STCAbs)}$$

$$\frac{\Gamma \vdash_c t_1[s]\ t_2[s] \overrightarrow{\in} T^*}{\Gamma \vdash_c (t_1\ t_2)[s] \overrightarrow{\in} T^*} \qquad \text{(STCApp)}$$

$$\frac{\begin{array}{c} \Gamma \vdash_c c_1 \overrightarrow{\in} T^* \\ \exists!\ T_1 \to T_2 \in T^* \mid \Gamma \vdash_c c_2 \overleftarrow{\in} T_1 \end{array}}{\Gamma \vdash_c c_1\ c_2 \overrightarrow{\in} \{T_2\} \Rightarrow (c_1 :: T_1 \to T_2)\ (c_2 :: T_1)} \qquad \text{(STCCApp)}$$

$$\boxed{\Gamma \vdash_c c \overleftarrow{\in} T \Rightarrow c}$$

$$\Gamma \vdash_c \mathsf{true} \overleftarrow{\in} \mathsf{Bool} \qquad \text{(CTCTrue)}$$

$$\Gamma \vdash_c \mathsf{false} \overleftarrow{\in} \mathsf{Bool} \qquad \text{(CTCFalse)}$$

$$\Gamma \vdash_c \mathsf{true}[s] \overleftarrow{\in} \mathsf{Bool} \qquad \text{(CTCTrue)}$$

$$\Gamma \vdash_c \mathsf{false}[s] \overleftarrow{\in} \mathsf{Bool} \qquad \text{(CTCFalse)}$$

$$\Gamma \vdash_c n \overleftarrow{\in} \mathsf{Int} \qquad \text{(CTCNum)}$$

$$\Gamma \vdash_c n[s] \overleftarrow{\in} \mathsf{Int} \qquad \text{(CTCCNum)}$$

$$\frac{\Gamma \vdash_c x[s] \overrightarrow{\in} \{T\}}{\Gamma \vdash_c x[s] \overleftarrow{\in} T} \qquad \text{(CTCVar}\Gamma\text{)}$$

$$\frac{x, \{(\overline{v : T})\} \in s \qquad T \in \{\overline{T}\}}{\Gamma \vdash_c x[s] \overleftarrow{\in} T \Rightarrow (x :: T)[s]} \qquad \text{(CTCVar}\phi\text{)}$$

$$\frac{\Gamma \vdash_c t[s] :: T \overleftarrow{\in} T}{\Gamma \vdash_c (t :: T)[s] \overleftarrow{\in} T} \qquad \text{(CTCAsc)}$$

$$\frac{\Gamma \vdash_c c \overleftarrow{\in} T}{\Gamma \vdash_c c :: T \overleftarrow{\in} \{T\}} \qquad \text{(CTCCAsc)}$$

$$\frac{\Gamma \vdash_c \mathsf{olet}\ x : T_1 = t_1[s]\ \mathsf{in}\ t_2[s] \overleftarrow{\in} T}{\Gamma \vdash_c (\mathsf{olet}\ x : T_1 = t_1\ \mathsf{in}\ t_2)[s] \overleftarrow{\in} T^*} \qquad \text{(CTCOLet)}$$

$$\frac{\begin{array}{c} x \notin dom(\Gamma) \qquad \Gamma \vdash_c c_1 \overrightarrow{\in} T_1^* \\ T_1 \in T_1^* \qquad \Gamma; \phi(s) \oplus (x : T_1) \vdash t_2 \overleftarrow{\in} T_2 \end{array}}{\Gamma \vdash_c \mathsf{olet}\ x : T_1 = c_1\ \mathsf{in}\ t_2[s] \overleftarrow{\in} T_2} \qquad \text{(CTCCOLet)}$$

$$\frac{x \notin dom(\Gamma \cup \phi(s)) \qquad \Gamma, x : T_1; \phi(s) \vdash t_2 \overleftarrow{\in} T_2}{\Gamma \vdash_c (\lambda x : T_1.\ t_2)[s] \overleftarrow{\in} T_1 \to T_2} \qquad \text{(CTCAbs)}$$

$$\frac{\Gamma \vdash_c t_1[s]\ t_2[s] \overleftarrow{\in} T}{\Gamma \vdash_c (t_1\ t_2)[s] \overleftarrow{\in} T} \qquad \text{(CTCApp)}$$

$$\frac{\begin{array}{c} \Gamma \vdash_c c_1 \overrightarrow{\in} T^* \\ \exists!\ T_1 \to T_2 \in T^* \mid \Gamma \vdash_c c_2 \overleftarrow{\in} T_1 \end{array}}{\Gamma \vdash_c c_1\ c_2 \overleftarrow{\in} T_2 \Rightarrow (c_1 :: T_1 \to T_2)\ (c_2 :: T_1)} \qquad \text{(CTCCApp)}$$

Figure 3: Configuration synthesis and checking.

$$\boxed{c \longrightarrow c}$$

$$\mathsf{true}[s] \longrightarrow \mathsf{true} \quad \text{(True)}$$

$$\mathsf{false}[s] \longrightarrow \mathsf{false} \quad \text{(False)}$$

$$n[s] \longrightarrow n \quad \text{(Num)}$$

$$x[x, \{(v : T_1)\} : s] \longrightarrow v \quad \text{(VarOk)}$$

$$\frac{x \neq y}{x[y \mapsto \{(\overline{v : T_1})\}, s] \longrightarrow x[s]} \quad \text{(VarNext)}$$

$$\frac{(v_i, T_i) \in \{(\overline{v : T})\}}{x[x \mapsto \{(\overline{v : T})\}, s] :: T_i \longrightarrow v_i} \quad \text{(VarAscOk)}$$

$$\frac{x \neq y}{x[y \mapsto \{(\overline{v : T_1})\}, s] :: T \longrightarrow x[s] :: T} \quad \text{(VarAscNext)}$$

$$(t :: T)[s] \longrightarrow t[s] :: T \quad \text{(AscSub)}$$

$$v :: T \longrightarrow v \quad \text{(Asc)}$$

$$\frac{c \longrightarrow c'}{c :: T \longrightarrow c' :: T} \quad \text{(Asc1)}$$

$$(\mathsf{olet}\ x : T_1 = t_1\ \mathsf{in}\ t_2)[s] \longrightarrow \mathsf{olet}\ x : T_1 = t_1[s]\ \mathsf{in}\ t_2[s] \quad \text{(LetSub)}$$

$$\mathsf{olet}\ x : T_1 = v\ \mathsf{in}\ t_2[s] \longrightarrow t_2[x, (v : T_1) :^* s] \quad \text{(Let)}$$

$$\frac{c_1 \longrightarrow c_1'}{\mathsf{olet}\ x : T_1 = c_1\ \mathsf{in}\ t_2[s] \longrightarrow \mathsf{olet}\ x : T_1 = c_1'\ \mathsf{in}\ t_2[s]} \quad \text{(Let1)}$$

$$(t_1\ t_2)[s] \longrightarrow t_1[s]\ t_2[s] \quad \text{(AppSub)}$$

$$(\lambda x : T_1.\ t_2)[s]\ v \longrightarrow ([x \mapsto v]t_2)[s] \quad \text{(App)}$$

$$\frac{c_1 \longrightarrow c_1'}{c_1\ c_2 \longrightarrow c_1'\ c_2} \quad \text{(App1)}$$

$$\frac{c \longrightarrow c'}{v\ c \longrightarrow v\ c'} \quad \text{(App2)}$$

Figure 4: Configuration reduction rules.

Proof. Immediate from the definition of the typing relation. □

Lemma 3 (Canonical Forms)**.**

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

Proof. Straightforward. □

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

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

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

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

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

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

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 $x \in dom(s)$, if the substitution $s = x \mapsto \{(\overline{v : T})\}, s'$, 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 $\Gamma \vdash_c c_1 : T_{11} \to T$, 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.

□

Definition 4 (Well typed substitution)**.** *A substitution s is said well typed with a typing context Γ, writing $\Gamma; \phi \vdash s$, if $dom(s) = dom(\Gamma)$ 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. □

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

Proof. By the definition of well typed substitution. □

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. □

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. □

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 $c = c_1\ c_2$, $\Gamma \vdash_c\ c_1 : T_2 \to T$ 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, $\Gamma \vdash_c c_1' : T_2 \to T$, 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 $\vdash c' : T$.

□