v	::=		values
		true	true value
		false	false value
		n	numeric value
		$\lambda x : T.\ t$	abstraction
nv	::=		non − values
		x	variable
		$t\ t$	application
		mlet $x : T = t$ in t	overloading let
		$t :: T$	ascription
		add1 t	sum
		not t	negation
t	::=		terms
		v	values
		nv	non − values
w	::=		configuration − values
		true	true value
		false	false value
		n	numeric value
		$(\lambda x : T.\ t)[s]$	abstraction

c	::=		configurations
		w	
		$t[s]$	
		$c :: T$	
		mlet $x : T = c$ in c	
		$c\ c$	
		add1 cc	
		not c	
		error	
T	::=		types
		Int	type of integers
		Bool	type of booleans
		$T \to T$	type of functions
Γ	::=		typing contexts
		$\varnothing$	empty context
		$\Gamma, x : T$	term variable binding
ϕ	::=		multi − typing contexts
		$\varnothing$	empty context
		$\phi, x : T^*$	term variable binding
s	::=		explicit substitutions
		$\bullet$	empty substitution
		$x \mapsto \{\overline{w}\}, s$	variable substitution

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

$\boxed{c \longrightarrow c}$

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

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

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

$$x[\,] \longrightarrow \mathsf{error} \qquad \text{(ErrVarFail)}$$

$$x[x \mapsto \{(\overline{w : T_1})\}, s \longrightarrow w_i \qquad \text{(VarOk)}$$

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

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

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

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

$$(\mathsf{add1}\ t_1)[s] \longrightarrow \mathsf{add1}\ t_1[s] \qquad \text{(SumSub)}$$

$$(\mathsf{not}\ t)[s] \longrightarrow \mathsf{not}\ t[s] \qquad \text{(NegationSub)}$$

$$w :: T \longrightarrow w \qquad \text{(Asc)}$$

$$\mathsf{mlet}\ x : T_1 = w\ \mathsf{in}\ t_2[s] \longrightarrow t_2[x \mapsto (w : T_1) \oplus s] \qquad \text{(Let)}$$

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

$$\mathsf{add1}\ w_1 \longrightarrow w_1 + 1 \qquad \text{(Sum)}$$

$$\mathsf{not}\ w \longrightarrow \neg\ w \qquad \text{(Negation)}$$

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

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

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

$$\frac{c \longrightarrow c'}{w\ c \longrightarrow w\ c'} \qquad \text{(App2)}$$

$$\frac{c_1 \longrightarrow c_1'}{\mathsf{add1}\ c_1 \longrightarrow \mathsf{add1}\ c_1'} \qquad \text{(Sum1)}$$

$$\frac{c \longrightarrow c'}{\mathsf{not}\ c \longrightarrow \mathsf{not}\ c'} \qquad \text{(Negation1)}$$

Figure 2: Configuration reduction rules.