$$\Gamma \vdash \texttt{true} : \texttt{bool} \tag{1}$$

$$\Gamma \vdash \texttt{false} : \texttt{bool} \tag{2}$$

$$\Gamma \vdash x : \Gamma(x) \tag{3}$$

$$\frac{\Gamma \vdash e_1 : \texttt{bool} \qquad \Gamma \vdash e_2 : \tau \qquad \Gamma \vdash e_3 : \tau}{\Gamma \vdash \texttt{if}\ e_1\ \texttt{then}\ e_2\ \texttt{else}\ e_3 : \tau} \tag{4}$$

$$\frac{\Gamma \cup \{(x_1, \tau_1)\} \vdash e : \tau_2}{\Gamma \vdash \texttt{fn}(x) \Rightarrow e : \tau_1 \to \tau_2} \tag{5}$$

$$\frac{\Gamma \vdash e_1 : \tau_1 \qquad \Gamma \vdash e_2 : \tau_1 \to \tau_2}{\Gamma \vdash e_1(e_2) : \tau_2} \tag{6}$$

We say that an expression e is *well-typed* in a type system if there exists an environment Γ and a type τ such that the judgment $\Gamma \vdash e : \tau$ can be proven by the type rules.

A type system is *sound* if well-typed programs cannot cause type errors.

What we want to do is prove BoolEm's type system to be sound.

An expression is *closed* if it has no free variables.

A *program* is a closed expression.

A *value* is a closed abstraction or a boolean constant. We'll use the value v to range over values.

$$(\texttt{fn}(x)\texttt{=>}e)(v) \longrightarrow e[v/x] \tag{7}$$

$$\texttt{if } v\texttt{then } e_2 \texttt{ else } e_3 \longrightarrow \begin{cases} e_2 & \text{if } v = \texttt{true} \\ e_3 & \text{otherwise} \end{cases} \tag{8}$$

$$\frac{e_1 \longrightarrow e_1'}{e_1(e_2) \longrightarrow e_1'(e_2)} \tag{9}$$

$$\frac{e_2 \longrightarrow e_2'}{v_1(e_2) \longrightarrow v_1(e_2')} \tag{10}$$

$$\frac{e_1 \longrightarrow e_1'}{\texttt{if } e_1 \texttt{ then } e_2 \texttt{ else } e_3 \longrightarrow \texttt{if } e_1' \texttt{ then } e_2 \texttt{ else } e_3} \tag{11}$$

An expression e is *stuck* if it is not a value and there does not exist an e' such that $e \longrightarrow e'$.

An expression *goes wrong* if it evaluates to a stuck expression.

$$\begin{tikzcd}
\text{Substitution (Lemma 1)} \arrow[d] & \text{Value Forms (Lemma 3)} \arrow[d] \\
\text{Type Preservation (Theorem 2)} \arrow[dr] & \text{Progress (Theorem 4)} \arrow[dl] \\
& \text{Soundness (Corollary 5)}
\end{tikzcd}$$

Lemma 1 (Substitution.) *If* $\Gamma \cup \{(x, \tau')\} \vdash e : \tau$ *and* $\Gamma \vdash v : \tau'$*, then* $\Gamma \vdash e[v/x] : \tau$*.*

Theorem 2 (Type Preservation.) *If* $\Gamma \vdash e : \tau$ *and* $e \longrightarrow e'$*, then* $\Gamma \vdash e' : \tau$*.*

Lemma 3 (Value Forms.) *If* $\Gamma \vdash v : bool$*, then* v *is in the form* `true` *or* `false`*. If* $\Gamma \vdash v : \tau_1 \to \tau_2$*, then* v *is in the form* $fn(x) \Rightarrow e$*.*

Theorem 4 (Progress.) *If* e *is a closed expression and* $\Gamma \vdash e : \tau$*, then either* e *is a value or there exists an* e' *such that* $e \longrightarrow e'$*.*

Corollary 5 (Soundness) *Well-typed programs cannot go wrong.*