Continuation

(Author: Hongwei Xi)

| Types     | $T ::= \ldots | cont(T_1, T_2)$ |
|------------|----------------------------------|
| Expressions| $e ::= \ldots | callcc(e) | throw(e_1, e_2)$ |
| Eval. ctx. | $E ::= \ldots | callcc(E) | throw(E, e)$ |
| Values     | $v ::= \ldots | E^*$ |

\[
\frac{x : T_1 \vdash E[x] : T_2 \quad \Gamma \vdash e : cont(T_1, T_2) \rightarrow T_1}{\Gamma \vdash E[callcc(e)] : T_2} \quad \text{(ty-callcc)}
\]

\[
\frac{x : T_0 \vdash E[x] : T_2 \quad \Gamma \vdash e_1 : cont(T_1, T_2) \quad \Gamma \vdash e_2 : T_1}{\Gamma \vdash E[throw(e_1, e_2)] : T_2} \quad \text{(ty-throw)}
\]

\[
\frac{x : T_1 \vdash E[x] : T_2}{\Gamma \vdash E^* : cont(T_1, T_2)} \quad \text{(ty-cont)}
\]

- $E[\text{callcc}(\lambda x.e)]$ is a redex, and its reduct is $E[e[x := E^*]]$.
- $E[\text{throw}(E^*_0, e)]$ is a redex, and its reduct is $E_0[e]$.

**Lemma 1 (Canonical Forms)** Assume that $\emptyset \vdash v : T$ is derivable.

- If $T = \text{cont}(T_1, T_2)$, then $v = E^*$ for some evaluation context $E$.

**Theorem 2 (Progress)** Assume that $\emptyset \vdash e : T$ is derivable. Then either $e$ is a value, or $e \rightarrow e'$ for some expression $e'$.

**Proof** By structural induction on $e$.

**Theorem 3 (Subject Reduction)** Assume that $\emptyset \vdash e : T$ is derivable and $e \rightarrow e'$. Then $\emptyset \vdash e' : T$ is derivable.

**Proof** By structural induction on the derivation $D$ of $\emptyset \vdash e : T$.

- The last applied rule in $D$ is (ty-callcc). Then $D$ is of the following form:

\[
\frac{x : T_1 \vdash E[x] : T_2 \quad \emptyset \vdash e_0 : cont(T_1, T_2) \rightarrow T_1}{\emptyset \vdash E[\text{callcc}(e_0)] : T_2} \quad \text{(ty-callcc)}
\]
where \( e = E[\text{callcc}(e_0)] \) and \( T = T_2 \).

If \( e_0 \) is not a value, then \( e_0 \to e'_0 \) for some \( e'_0 \) and \( e' = \text{callcc}(e'_0) \). By induction hypothesis on \( D_1 \), we know that \( \emptyset \vdash e'_0 : \text{cont}(T_1, T_2) \) is derivable. Hence, \( \emptyset \vdash e' : T_2 \) can be derived as follows:

\[
\begin{array}{c}
x : T_1 \vdash E[x] : T_2 \quad D_1 :: \emptyset \vdash e'_0 : \text{cont}(T_1, T_2) \to T_1 \\
\emptyset \vdash E[\text{callcc}(e'_0)] : T_2
\end{array}
\] (ty-callcc)

We now assume that \( e_0 \) is a value. By the lemma of canonical forms, \( e_0 \) must be of the form \( \lambda x.e_1 \). Hence, \( e' = E[e_1[x := E^*]] \). Note that \( \emptyset \vdash E^* : \text{cont}(T_1, T_2) \) is derivable. By the substitution lemma, we know that \( \emptyset \vdash e_1[x := E^*] : T_1 \) is derivable. Hence, \( \emptyset \vdash e' : T_2 \) is also derivable.

- The last applied rule in \( D \) is (ty-throw). Then \( D \) is of the following form:

\[
\begin{array}{c}
x : T_0 \vdash E[x] : T_2 \quad D_1 :: \Gamma \vdash e_1 : \text{cont}(T_1, T_2) \quad D_2 :: \Gamma \vdash e_2 : T_1 \\
\Gamma \vdash E[\text{throw}(e_1, e_2)] : T_2
\end{array}
\] (ty-throw)

where \( e = E[\text{throw}(e_1, e_2)] \) and \( T = T_2 \).

If \( e_1 \) is not a value, then \( e_1 \to e'_1 \) holds for some \( e'_1 \) and \( e' = \text{throw}(e'_1, e_2) \). By induction hypothesis on \( D_1 \), \( \emptyset \vdash e'_1 : \text{cont}(T_1, T_2) \) is derivable. Hence, we can derive \( \emptyset \vdash e' : T \) as follows:

\[
\begin{array}{c}
x : T_0 \vdash E[x] : T_2 \quad D_1 :: \Gamma \vdash e'_1 : \text{cont}(T_1, T_2) \quad D_2 :: \Gamma \vdash e_2 : T_1 \\
\Gamma \vdash E[\text{throw}(e'_1, e_2)] : T_2
\end{array}
\] (ty-throw)

We now assume that \( e_1 \) is a value. By the lemma of canonical forms, \( e_1 \) must be of the form \( E_0^* \) for some evaluation context \( E_0 \). Hence, \( e' = E_0[\text{throw}(e'_1, e_2)] \). Clearly, \( x : T_1 \vdash E_0[x] : T_2 \) is derivable, which implies that \( \emptyset \vdash e' : T \) is also derivable.

The other cases have all been handled before.