Queens' College Cambridge

Types

Alistair O'Brien

Department of Computer Science

June 2, 2022

1 Simply-Typed λ-Calculus

Syntax

Types	$\tau ::= 1 \mid 0 \mid \tau \to \tau \mid \tau \times \tau \mid \tau + \tau$
Terms	$e ::= x \mid \langle\rangle \mid \langle e_1, e_2\rangle \mid \mathsf{fst}\ e \mid \mathsf{snd}\ e \mid \mathsf{L}\ e \mid \mathsf{R}\ e$ $\mid \mathsf{case}(e, \mathsf{L}\ x \to e, \mathsf{R}\ x \to e) \mid \lambda x : \tau.e \mid e\ e \mid \mathsf{abort}\ e$
Values	$v ::= \langle\rangle \mid \langle v_1, v_2\rangle \mid \lambda x : \tau.e \mid \mathsf{L}\ v \mid \mathsf{R}\ v$
Contexts	$\Gamma ::= \cdot \mid \Gamma, x : \tau$
Evaluation Contexts	$\mathbb{E} ::= [\cdot] \mid \langle \mathbb{E}, e\rangle \mid \langle v, \mathbb{E}\rangle \mid \mathsf{fst}\ \mathbb{E} \mid \mathsf{snd}\ \mathbb{E} \mid \mathsf{L}\ \mathbb{E} \mid \mathsf{R}\ \mathbb{E}$ $\mid \mathsf{case}(\mathbb{E}, \mathsf{L}\ x \to e, \mathsf{R}\ x \to e) \mid \mathbb{E}\ e \mid v\ \mathbb{E} \mid \mathsf{abort}\ \mathbb{E}$

Typing Rules

(I: introduction rule, E: elimination rule, Hyp: hypothesis)

$$\frac{x : \tau \in \Gamma}{\Gamma \vdash x : \tau}\ \textsc{Hyp} \qquad \frac{}{\Gamma \vdash \langle\rangle : 1}\ 1\mathrm{I} \qquad (\textsc{No elimination for } 1)$$

$$(\textsc{No introduction for } 0) \qquad \frac{\Gamma \vdash e : 0}{\Gamma \vdash \mathsf{abort}\ e : \tau}\ 0\mathrm{E} \qquad \frac{\Gamma \vdash e_1 : \tau_1 \qquad \Gamma \vdash e_2 : \tau_2}{\Gamma \vdash \langle e_1, e_2\rangle : \tau_1 \times \tau_2}\ {\times}\mathrm{I}$$

$$\frac{\Gamma \vdash e : \tau_1 \times \tau_2}{\Gamma \vdash \mathsf{fst}\ e : \tau_1}\ {\times}\mathrm{E}_1 \qquad \frac{\Gamma \vdash e : \tau_1 \times \tau_2}{\Gamma \vdash \mathsf{snd}\ e : \tau_2}\ {\times}\mathrm{E}_2 \qquad \frac{\Gamma, x : \tau_1 \vdash e : \tau_2}{\Gamma \vdash \lambda x : \tau_1.e : \tau_1 \to \tau_2}\ {\to}\mathrm{I}$$

$$\frac{\Gamma \vdash e_1 : \tau_1 \to \tau_2 \qquad \Gamma \vdash e_2 : \tau_1}{\Gamma \vdash e_1\ e_2 : \tau_2}\ {\to}\mathrm{E} \qquad \frac{\Gamma \vdash e : \tau_1}{\Gamma \vdash \mathsf{L}\ e : \tau_1 + \tau_2}\ {+}\mathrm{I}_1 \qquad \frac{\Gamma \vdash e : \tau_2}{\Gamma \vdash \mathsf{R}\ e : \tau_1 + \tau_2}\ {+}\mathrm{I}_2$$

$$\frac{\Gamma \vdash e : \tau_1 + \tau_2 \qquad \Gamma, x_1 : \tau_1 \vdash e_1 : \tau \qquad \Gamma, x_2 : \tau_2 \vdash e_2 : \tau}{\Gamma \vdash \mathsf{case}(e, \mathsf{L}\ x_1 \to e_1, \mathsf{R}\ x_2 \to e_2) : \tau}\ {+}\mathrm{E}$$

Operational Semantics

(Red: reduction rule (intro form *immediately* followed by elim form),
Eval: acts recursively on a subterm, controls *evaluation order* using *evaluation context*)

(No rule for unit) $\dfrac{e \rightsquigarrow e'}{\mathbb{E}[e] \rightsquigarrow \mathbb{E}[e']}$ EvalCtx $\dfrac{}{\mathsf{fst}\ \langle v_1, v_2\rangle \rightsquigarrow v_1}$ RedFst

$\dfrac{}{\mathsf{snd}\ \langle v_1, v_2\rangle \rightsquigarrow v_2}$ RedSnd $\dfrac{}{\mathsf{case}(\mathsf{L}\ v, \mathsf{L}\ x_1 \to e_1, \mathsf{R}\ x_2 \to e_2) \rightsquigarrow \{v/x_1\}e_1}$ RedCase$_1$

$$\dfrac{}{\mathsf{case}(\mathsf{R}\ v, \mathsf{L}\ x_1 \to e_1, \mathsf{R}\ x_2 \to e_2) \rightsquigarrow \{v/x_2\}e_2}\ \textsc{RedCase}_2$$

$$\dfrac{}{(\lambda x : \tau.e)\ v \rightsquigarrow \{v/x\}e}\ \textsc{RedFn}$$

Theorems

Lemma 1.0.1. (**Weakening and Exchange**)

(i) If $\Gamma_1, \Gamma_2 \vdash e : \tau$, then $\Gamma_1, x : \tau_x, \Gamma_2 \vdash e : \tau$.

(ii) If $\Gamma_1, x_1 : \tau_1, x_2 : \tau_2, \Gamma_2 \vdash e : \tau$, then $\Gamma_1, x_2 : \tau_2, x_1 : \tau_1, \Gamma_2 \vdash e : \tau$.

Proof. Proof by *rule induction* on the premises.

Theorem 1.0.1. (**Substitution**) If $\Gamma \vdash e : \tau_1$ and $\Gamma, x : \tau_1 \vdash e_2 : \tau_2$ holds, then $\Gamma \vdash \{e_1/x\}e_2 : \tau_2$ holds.

Proof. Proof by *rule induction* on $\Gamma, x : \tau_1 \vdash e_2 : \tau_2$

Theorem 1.0.2. (**Progress**) If $\cdot \vdash e : \tau$, then e is a value or $\exists e' \in \Lambda. e \rightsquigarrow e'$.

Proof. Proof by *rule induction* on $\cdot \vdash e : \tau$.

Theorem 1.0.3. (**Preservation**) If $\cdot \vdash e : \tau$ and $e \rightsquigarrow e'$, then $\cdot \vdash e' : \tau$.

Proof. Proof by *rule induction* on $e \rightsquigarrow e'$.

- **Safety = Progress + Preservation**

Theorem 1.0.4. (**Determinacy**) If $e_1 \rightsquigarrow e_2$ and $e_1 \rightsquigarrow e_3$, then $e_2 = e_3$.

Proof. Proof by *rule induction* on $e_1 \rightsquigarrow e_2$.

Termination

Definition 1.0.1. (**Halting**) A term e is said to half if and only if there exists a value v s.t $e \rightsquigarrow^* v$.

Definition 1.0.2. (**Type Interpretation**) The interpretation of the type τ is defined by the denotation $[\![\tau]\!] \subseteq \Lambda$:

$$\begin{aligned}
[\![0]\!] &= \emptyset \\
[\![1]\!] &= \{e \in \Lambda : e \text{ halts}\} \\
[\![\tau_1 \to \tau_2]\!] &= \{e \in \Lambda : e \text{ halts} \wedge \forall e' \in \Lambda. e' \in [\![\tau_1]\!] \implies e\ e' \in [\![\tau_2]\!]\} \\
[\![\tau_1 \times \tau_2]\!] &= \{e \in \Lambda : e \text{ halts} \wedge \mathsf{fst}\ e \in [\![\tau_1]\!] \wedge \mathsf{snd}\ e \in [\![\tau_2]\!]\} \\
[\![\tau_1 + \tau_2]\!] &= \{e \in \Lambda : e \text{ halts} \\
&\qquad \wedge \forall f_1, f_2 \in \Lambda, \tau \in \mathsf{Type}. f_i \in [\![\tau_i \to \tau]\!] \implies \mathsf{case}(e, \mathsf{L}\ x_1 \to f_1\ x_1, \mathsf{R}\ x_2 \to f_2\ x_2) \in [\![\tau]\!]\}
\end{aligned}$$

Lemma 1.0.2. (**Closure**) If $e \rightsquigarrow e'$, then $e' \in [\![\tau]\!] \iff e' \in [\![\tau]\!]$.

Proof. Proof by *structural induction* on τ.

Definition 1.0.3. (**Context Interpretations**) The interpretation of a term context Γ is defined by the denotation $[\![\Gamma\ \mathsf{ctx}]\!] \subseteq \mathsf{TermSubst}$ is

$$\begin{aligned}[\![\cdot\ \mathsf{ctx}]\!] &= \{\cdot\} \\ [\![\Gamma, x : \tau\ \mathsf{ctx}]\!] &= \{\{\phi, e/x\} \in \mathsf{TermSubst} : \phi \in [\![\Gamma\ \mathsf{ctx}]\!] \wedge e \in [\![\tau]\!]\}\end{aligned}$$

Lemma 1.0.3. (**The Fundamental Lemma**) If $\Gamma \vdash e : \tau$, then for all $\phi \in [\![\Gamma\ \mathsf{ctx}]\!]$, $\phi e \in [\![\tau]\!]$.

Proof. Proof by *rule induction* on $\Gamma \vdash e : \tau$.

Theorem 1.0.5. (**Consistency**) For all $e \in \Lambda$. $\cdot \nvdash e : 0$.

Proof. Let $e \in \Lambda$ be arbitrary. We proceed by contradiction. Assume $\cdot \vdash e : 0$ holds. By the fundamental lemma, $e \in [\![\tau]\!] = \emptyset$. $e \notin \emptyset$ by the definable property of $\emptyset$. A contradiction!

2 Polymorphic λ-Calculus (System F)

Syntax

Types	$A ::= \alpha \mid A \to A \mid \forall\alpha.A \mid \exists\alpha.A$
Terms	$e ::= x \mid \lambda x : A.e \mid e\ e \mid \Lambda\alpha.e \mid e\ [A] \mid \mathsf{pack}_{\alpha_{\text{abs}}.A_{\text{sig}}}(A_{\text{conc}}, e_{\text{impl}})$
	$\mid \mathsf{let\ pack}(\alpha, x) = e \mathsf{\ in\ } e$
Values	$v ::= \lambda x : A.e \mid \Lambda\alpha.e \mid \mathsf{pack}_{\alpha.A}(A, v)$
Type Contexts	$\Theta ::= \cdot \mid \Theta, \alpha$
Term Contexts	$\Gamma ::= \cdot \mid \Gamma, x : A$
Evaluation Contexts	$\mathbb{E} ::= [\cdot] \mid \mathbb{E}\ e \mid v\ \mathbb{E} \mid \mathbb{E}\ [A] \mid \mathsf{pack}_{\alpha.A}(A, \mathbb{E}) \mid \mathsf{let\ pack}(\alpha, x) = \mathbb{E} \mathsf{\ in\ } e$

Typing Rules

$\boxed{\Theta \vdash A \ \mathsf{type}}$

$$\frac{\alpha \in \Theta}{\Theta \vdash \alpha \ \mathsf{type}} \qquad \frac{\Theta \vdash A_1 \ \mathsf{type} \qquad \Theta \vdash A_2 \ \mathsf{type}}{\Theta \vdash A_1 \to A_2 \ \mathsf{type}} \qquad \frac{\Theta, \alpha \vdash A \ \mathsf{type}}{\Theta \vdash \forall\alpha.A \ \mathsf{type}} \qquad \frac{\Theta, \alpha \vdash A \ \mathsf{type}}{\Theta \vdash \exists\alpha.A \ \mathsf{type}}$$

$\boxed{\Theta \vdash \Gamma \ \mathsf{ctx}}$

$$\frac{}{\Theta \vdash \cdot \ \mathsf{ctx}} \qquad \frac{\Theta \vdash \Gamma \ \mathsf{ctx} \qquad x \notin \mathrm{dom}\ \Gamma \qquad \Theta \vdash A \ \mathsf{type}}{\Theta \vdash \Gamma, x : A \ \mathsf{ctx}}$$

$\boxed{\Theta; \Gamma \vdash e : A}$

$$\frac{x : A \in \Gamma}{\Theta; \Gamma \vdash x : A}\ \textsc{Hyp} \qquad \frac{\Theta \vdash A \ \mathsf{type} \qquad \Theta; \Gamma, x : A_1 \vdash e : A_2}{\Theta; \Gamma \vdash \lambda x : A_1.e : A_1 \to A_2}\ {\to}\mathrm{I}$$

$$\frac{\Theta; \Gamma \vdash e_1 : A_1 \to A_2 \qquad \Theta; \Gamma \vdash e_2 : A_1}{\Theta; \Gamma \vdash e_1\ e_2 : A_2}\ {\to}\mathrm{E} \qquad \frac{\Theta, \alpha; \Gamma \vdash e : A}{\Theta; \Gamma \vdash \Lambda\alpha.e : \forall\alpha.A}\ \forall\mathrm{I}$$

$$\frac{\Theta; \Gamma \vdash e : \forall\alpha.A_1 \qquad \Theta \vdash A_2 \ \mathsf{type}}{\Theta; \Gamma \vdash e\ [A_2] : \{A_2/\alpha\}A_1}\ \forall\mathrm{E}$$

$$\frac{\Theta; \alpha_{\text{abs}} \vdash A_{\text{sig}} \ \mathsf{type} \qquad \Theta \vdash A_{\text{conc}} \ \mathsf{type} \qquad \Theta; \Gamma \vdash e_{\text{impl}} : \{A_{\text{conc}}/\alpha_{\text{abs}}\}A_{\text{sig}}}{\Theta; \Gamma \vdash \mathsf{pack}_{\alpha_{\text{abs}}.A_{\text{sig}}}(A_{\text{conc}}, e_{\text{impl}}) : \exists \alpha_{\text{abs}}.A_{\text{sig}}} \ \exists\text{I}$$

$$\frac{\Theta; \Gamma \vdash e_1 : \exists \alpha_{\text{abs}}.A_{\text{sig}} \qquad \Theta, \alpha; \Gamma, x : \{\alpha_{\text{abs}}/\alpha\}A_{\text{sig}} \vdash e_2 : A}{\Theta; \Gamma \vdash \mathsf{let\ pack}(\alpha, x) = e_1 \ \mathsf{in} \ e_2 : A} \ \exists\text{E}$$

Operational Semantics

$$\frac{e \rightsquigarrow e'}{\mathbb{E}[e] \rightsquigarrow \mathbb{E}[e']} \ \textsc{EvalCtx} \qquad \frac{}{(\lambda x : A.e) \ v \rightsquigarrow \{v/x\}e} \ \textsc{RedFn}$$

$$\frac{}{(\lambda \alpha.e) \ [A] \rightsquigarrow \{A/\alpha\}e} \ \textsc{RedForall}$$

$$\frac{}{\mathsf{let\ pack}(\alpha, x) = \mathsf{pack}_{\alpha_{\text{abs}}.A_{\text{sig}}}(A_{\text{conc}}, v_{\text{impl}}) \ \mathsf{in} \ e \rightsquigarrow \{A_{\text{conc}}/\alpha, v_{\text{impl}}/x\}e} \ \textsc{RedExists}$$

Church Encodings

Pairs

$$\begin{aligned}
A_1 \times A_2 &\triangleq \forall \alpha.\ (A_1 \to A_2 \to \alpha) \to \alpha \\
\langle e_1, e_2 \rangle &\triangleq \Lambda \alpha.\ \lambda k : A_1 \to A_2 \to \alpha.\ k\ e\ e' \\
\mathsf{fst}\ e &\triangleq e\ A_1\ (\lambda x : A_1.\ \lambda y : A_2.\ x) \\
\mathsf{snd}\ e &\triangleq e\ A_2\ (\lambda x : A_1.\ \lambda y : A_2.\ y)
\end{aligned}$$

Sums

$$\begin{aligned}
A_1 + A_2 &\triangleq \forall \alpha.\ (A_1 \to \alpha) \to (A_2 \to \alpha) \to \alpha \\
\mathsf{L}\ e &\triangleq \Lambda \alpha.\ \lambda f_1 : A_1 \to \alpha.\ \lambda f_2 : A_2 \to \alpha.\ f_1\ e \\
\mathsf{R}\ e &\triangleq \Lambda \alpha.\ \lambda f_1 : A_1 \to \alpha.\ \lambda f_2 : A_2 \to \alpha.\ f_2\ e
\end{aligned}$$

$$\mathsf{case}(e, \mathsf{L}\ x_1 \to e_1, \mathsf{R}\ x_2 \to e_2) : A \ \triangleq \ e\ [A]\ (\lambda x_1 : A_1.\ e_1)\ (\lambda x_2 : A_2.\ e_2)$$

Existential types

$$\exists \alpha.\ A_{\text{sig}} \ \triangleq \ \forall \beta.\ (\forall \alpha.\ A_{\text{sig}} \to \beta) \to \beta$$

$$\mathsf{pack}_{\alpha_{\text{abs}}.A_{\text{sig}}}(A_{\text{conc}}, e_{\text{impl}}) \ \triangleq \ \Lambda \beta.\ \lambda k : \forall \alpha_{\text{abs}}.\ A_{\text{sig}} \to \beta.\ k\ [A_{\text{conc}}]\ e_{\text{impl}}$$

$$\mathsf{let\ pack}(\alpha, x) = e_{\text{impl}}\ \mathsf{in}\ e : A \ \triangleq \ e_{\text{impl}}\ [A]\ (\Lambda \alpha.\ \lambda x : A_{\text{sig}}.\ e)$$

Booleans

$$\begin{aligned}
\mathsf{bool} &\triangleq \forall\alpha.\ \alpha \to \alpha \to \alpha \\
\mathsf{True} &\triangleq \Lambda\alpha.\ \lambda x : \alpha.\ \lambda y : \alpha.\ x \\
\mathsf{False} &\triangleq \Lambda\alpha.\ \lambda x : \alpha.\ \lambda y : \alpha.\ y \\
\mathsf{if}\ e\ \mathsf{then}\ e_1\ \mathsf{else}\ e_2 : A &\triangleq e\ [A]\ e_1\ e_2
\end{aligned}$$

Natural numbers

$$\begin{aligned}
\mathbb{N} &\triangleq \forall\alpha.\ \alpha \to (\alpha \to \alpha) \to \alpha \\
\mathsf{zero} &\triangleq \Lambda\alpha.\ \lambda z : \alpha.\ \lambda s : \alpha \to \alpha.\ z \\
\mathsf{succ}(e) &\triangleq \Lambda\alpha.\ \lambda z : \alpha.\ \lambda s : \alpha \to \alpha.\ s\ (e\ [\alpha]\ z\ s) \\
\mathsf{iter}(e, \mathsf{zero} \to e_{\mathsf{z}}, \mathsf{succ}(x) \to e_{\mathsf{s}}) : [A] &\triangleq e\ [A]\ e_{\mathsf{z}}\ (\lambda x : A.\ e_{\mathsf{s}})
\end{aligned}$$

Lists

$$\begin{aligned}
\mathsf{list}\ A &\triangleq \forall\alpha.\ \alpha \to (A \to \alpha \to \alpha) \to \alpha \\
[] &\triangleq \Lambda\alpha.\ \lambda n : \alpha.\ \lambda c : A \to \alpha \to \alpha.\ n \\
e :: e' &\triangleq \Lambda\alpha.\ \lambda n : \alpha.\ \lambda c : A \to \alpha \to \alpha.\ c\ e\ (e'\ [\alpha]\ n\ c) \\
\mathsf{fold}(e, [] \to e_{\mathsf{n}}, x :: r \to e_{\mathsf{c}}) : B &\triangleq e\ [B]\ e_{\mathsf{n}}\ (\lambda x : A.\ \lambda r : B.\ e_{\mathsf{c}})
\end{aligned}$$

Theorems

Lemma 2.0.1. (**Type Weakening, Exchange and Substitution**)

(i) If $\Theta_1, \Theta_2 \vdash A\ \mathsf{type}$, then $\Theta_1, \beta, \Theta_2 \vdash A\ \mathsf{type}$.

(ii) If $\Theta_1, \alpha_1, \alpha_2, \Theta_2 \vdash A\ \mathsf{type}$, then $\Theta_1, \alpha_2, \alpha_1, \Theta_2 \vdash A\ \mathsf{type}$.

(iii) If $\Theta \vdash A\ \mathsf{type}$ and $\Theta, \alpha \vdash B\ \mathsf{type}$, then $\Theta \vdash \{A/\alpha\}B\ \mathsf{type}$.

Proof. Proof by *rule induction* on $\Theta_1, \Theta_2 \vdash A\ \mathsf{type}$, $\Theta_1, \alpha_1, \alpha_2, \Theta_2 \vdash A\ \mathsf{type}$ and $\Theta, \alpha \vdash B\ \mathsf{type}$.

Lemma 2.0.2. (**Context Weakening, Exchange and Substitution**)

(i) If $\Theta_1, \Theta_2 \vdash \Gamma\ \mathsf{ctx}$, then $\Theta_1, \alpha, \Theta_2 \vdash \Gamma\ \mathsf{ctx}$.

(ii) If $\Theta_1, \alpha_1, \alpha_2, \Theta_2 \vdash \Gamma\ \mathsf{ctx}$, then $\Theta_1, \alpha_2, \alpha_1, \Theta_2 \vdash \Gamma\ \mathsf{ctx}$.

(iii) If $\Theta \vdash A\ \mathsf{type}$ and $\Theta, \alpha \vdash \Gamma\ \mathsf{ctx}$, then $\Theta \vdash \{A/\alpha\}\Gamma\ \mathsf{ctx}$.

Proof. Proof by *rule induction* on $\Theta \vdash \Gamma \mathsf{ctx}$. *Lifts type-level structural properties to contexts.*

Lemma 2.0.3. (**Regularity**) If $\Theta \vdash \Gamma\ \mathsf{ctx}$ and $\Theta; \Gamma \vdash e : A$, then $\Theta \vdash A\ \mathsf{type}$.

Proof. Proof by *rule induction* on $\Theta; \Gamma \vdash e : A$.

Lemma 2.0.4. (**Type Weakening, Exchange and Substitution of Terms**)

(i) If $\Theta_1, \Theta_2 \vdash \Gamma$ ctx and $\Theta_1, \Theta_2; \Gamma \vdash e : A$, then $\Theta_1, \alpha, \Theta_2; \Gamma \vdash e : A$.

(ii) If $\Theta_1, \alpha_1, \alpha_2, \Theta_2 \vdash \Gamma$ ctx and $\Theta_1, \alpha_1, \alpha_2, \Theta_2; \Gamma \vdash e : A$, then $\Theta_1, \alpha_2, \alpha_1, \Theta_2; \Gamma \vdash e : A$.

(iii) If $\Theta, \alpha \vdash \Gamma$ ctx and $\Theta \vdash A$ type and $\Theta, \alpha; \Gamma \vdash e : B$, then $\Theta; \{A/\alpha\}\Gamma \vdash \{A/\alpha\}e : \{A/\alpha\}B$.

Proof. Proof by *rule induction* on $\Theta_1, \Theta_2; \Gamma \vdash e : A$, $\Theta_1, \alpha_1, \alpha_2, \Theta_2; \Gamma \vdash e : A$ and $\Theta, \alpha; \Gamma \vdash e : B$.

Lemma 2.0.5. (**Context Weakening, Exchange and Substitution of Terms**)

(i) If $\Theta \vdash \Gamma_1, \Gamma_2$ ctx and $\Theta \vdash A_x$ type and $\Theta; \Gamma_1, \Gamma_2 \vdash e : A$, then $\Theta; \Gamma_1, x : A_x, \Gamma_2 \vdash e : A$.

(ii) If $\Theta \vdash \Gamma_1, x_1 : A_1, x_2 : A_2, \Gamma_2$ ctx and $\Theta; \Gamma_1, x_1 : A_1, x_2 : A_2, \Gamma_2 \vdash e : A$, then $\Theta; \Gamma_1, x_2 : A_2, x_1 : A_1, \Gamma_2 \vdash e : A$.

(iii) If $\Theta \vdash \Gamma, x : A$ ctx and $\Theta; \Gamma \vdash e_1 : A_1$ and $\Theta; \Gamma, x : A_1 \vdash e_2 : A_2$, then $\Theta; \Gamma \vdash \{e_1/x\}e_2 : A_2$.

Proof. Proof by *rule induction* on $\Theta; \Gamma_1, \Gamma_2 \vdash e : A$, $\Theta; \Gamma_1, x_1 : A_1, x_2 : A_2, \Gamma_2 \vdash e : A$ and $\Theta; \Gamma, x : A_1 \vdash e_2 : A_2$.

Theorem 2.0.1. (**Progress**) If $\cdot; \cdot \vdash e : A$, then e is a value or $\exists e' \in \Lambda. e \rightsquigarrow e'$.

Proof. Proof by *rule induction* on $\cdot; \cdot \vdash e : A$.

Theorem 2.0.2. (**Preservation**) If $\cdot; \cdot \vdash e : A$ and $e \rightsquigarrow e'$, then $\cdot; \cdot \vdash e' : A$.

Proof. Proof by *rule induction* on $e \rightsquigarrow e'$.

Termination

Definition 2.0.1. (**Semantic Type**) A semantic type is a set of closed terms $X \in \mathsf{SemType}$ s.t

(i) (Halting). If $e \in X$ then e halts.

(ii) (Closure). If $e \rightsquigarrow e'$, then $e \in X \iff e' \in X$.

Definition 2.0.2. (**Type Assignment**) A *type assignment* is a partial function $\theta : \mathsf{TypeVar} \rightharpoonup \mathsf{Type}$. θ is said to be a Θ-assignment, written $\theta \in \mathsf{Assign}(\Theta)$, if $\mathrm{dom}\ \theta = \mathrm{dom}\ \Theta$.

Definition 2.0.3. (**Type Interpretation**) The interpretation of types is defined by the denotation $[\![\Theta \vdash A \ \mathsf{type}]\!] : \mathsf{TypeAssign} \to \mathsf{SemType}$:

$$
\begin{aligned}
[\![\Theta \vdash \alpha \ \mathsf{type}]\!]\,\theta &= \theta(\alpha) \\
[\![\Theta \vdash A \to B \ \mathsf{type}]\!]\,\theta &= \{e \in \Lambda : e \text{ halts} \wedge \forall e' \in [\![\Theta \vdash A \ \mathsf{type}]\!]\,\theta . e\ e' \in [\![\Theta \vdash B \ \mathsf{type}]\!]\,\theta\} \\
[\![\Theta \vdash \forall \alpha . B]\!]\,\theta &= \{e \in \Lambda : e \text{ halts} \\
&\qquad \wedge \forall A \in \mathsf{Type}, X \in \mathsf{SemType}. e\ [A] \in [\![\Theta, \alpha \vdash B \ \mathsf{type}]\!]\,(\theta, X/\alpha)\}
\end{aligned}
$$

Lemma 2.0.6. (**Closure**) If $\theta \in \mathsf{Assign}(\Theta)$, then $[\![\Theta \vdash A \ \mathsf{type}]\!]\,\theta \in \mathsf{SemType}$.

Proof. Proof by *rule induction* on $\Theta \vdash A$ type.

Lemma 2.0.7. (**Exchange and Weakening**)

(i) $[\![\Theta_1, \alpha_1, \alpha_2, \Theta_2 \vdash A \text{ type}]\!] = [\![\Theta_1, \alpha_2, \alpha_1, \Theta_2 \vdash A \text{ type}]\!]$

(ii) If $\Theta \vdash A$ type, then $[\![\Theta, \alpha \vdash A \text{ type}]\!]\,(\theta, X/\alpha) = [\![\Theta \vdash A \text{ type}]\!]\,\theta$

Proof. Proof by *rule induction* on $\Theta_1, \alpha_1, \alpha_2, \Theta \vdash A$ type and $\Theta, \alpha \vdash A$ type.

Lemma 2.0.8. (**Substitution**) If $\Theta \vdash A$ type and $\Theta, \alpha \vdash B$ type, then $[\![\Theta \vdash \{A/\alpha\}B \text{ type}]\!]\,\theta = [\![\Theta, \alpha \vdash B \text{ type}]\!]\,(\theta, [\![\Theta \vdash A \text{ type}]\!]\,\theta)$.

Proof. Proof by *rule induction* on $\Theta, \alpha \vdash B$ type.

Definition 2.0.4. (**Context Interpretations**) The interpretation of a term context Γ is defined by the denotation $[\![\Theta \vdash \Gamma \text{ ctx}]\!] : \mathsf{TypeAssign} \to \mathcal{P}(\mathsf{TermSubst})$ is

$$\begin{aligned}
[\![\Theta \vdash \cdot \text{ ctx}]\!]\,\theta &= \{\cdot\} \\
[\![\Theta \vdash \Gamma, x : A \text{ ctx}]\!] &= \{\{\phi, e/x\} \in \mathsf{TermSubst} : \phi \in [\![\Theta \vdash \Gamma \text{ ctx}]\!]\,\theta \wedge [\![\Theta \vdash A \text{ type}]\!]\,\theta\}
\end{aligned}$$

The interpretation of the type context Θ is defined by the denotation $[\![\Theta \text{ ctx}]\!] \subseteq \mathsf{TypeSubst}$:

$$\begin{aligned}
[\![\cdot \text{ ctx}]\!] &= \{\cdot\} \\
[\![\Theta, \alpha \text{ ctx}]\!] &= \{\{\varphi, A/\alpha\} \in \mathsf{TypeSubst} : \varphi \in [\![\Theta \text{ ctx}]\!] \wedge A \in \mathsf{Type}\}
\end{aligned}$$

Lemma 2.0.9. (**The Fundamental Lemma**) If $\Theta; \Gamma \vdash e : A$ and $\Theta \vdash \Gamma$ ctx, then for all $\theta \in \mathsf{Assign}(\Theta)$, $\phi \in [\![\Theta \vdash \Gamma \text{ ctx}]\!]\,\theta$ and $\varphi \in [\![\Theta \text{ ctx}]\!]$, $\varphi\phi e \in [\![\Theta \vdash A \text{ type}]\!]\,\theta$ holds.

Proof. Proof by *rule induction* on $\Theta; \Gamma \vdash e : A$.

3 Monadic Effects

Syntax

Types	$\tau ::= 1 \mid \mathbb{N} \mid \tau \to \tau \mid \mathsf{ref}\ \tau \mid \mathsf{state}\ \tau$
Pure Terms	$e ::= x \mid \langle\rangle \mid n \mid \lambda x : \tau.e \mid e\ e \mid \ell \mid \{t\}$
Impure Terms	$t ::= \mathsf{ref}\ e \mid !e \mid e := e' \mid \mathsf{let}\ x = e; t \mid \mathsf{return}\ e$
Values	$v ::= \langle\rangle \mid n \mid \lambda x : \tau.e \mid \ell \mid \{t\}$
Contexts	$\Gamma ::= \cdot \mid \Gamma, x : \tau$
Store	$\sigma ::= \cdot \mid \sigma, \ell \mapsto v$
Store Context	$\Sigma ::= \cdot \mid \Sigma, \ell : \tau$
Pure Evaluation Contexts	$\mathbb{E} ::= [\cdot] \mid \mathbb{E}\ e \mid v\ \mathbb{E}$
Impure Evaluation Contexts	$\mathbb{T} ::= \mathsf{ref}\ \mathbb{E} \mid !\mathbb{E} \mid \mathbb{E} := e \mid v := \mathbb{E} \mid \mathsf{let}\ x = \mathbb{E}; t$ $\mid \mathsf{return}\ \mathbb{E}$

Typing Rules

$\boxed{\Sigma; \Gamma \vdash e : \tau}$

$$\frac{x : \tau \in \Gamma}{\Sigma; \Gamma \vdash x : \tau}\ \textsc{Hyp} \qquad \frac{}{\Sigma; \Gamma \vdash e : 1}\ 1\mathrm{I} \qquad \frac{}{\Sigma; \Gamma \vdash n : \mathbb{N}}\ \mathbb{N}\mathrm{I} \qquad \frac{\Sigma; \Gamma, x : \tau_1 \vdash e : \tau_2}{\Sigma; \Gamma \vdash \lambda x : \tau_1.e : \tau_2}\ {\to}\mathrm{I}$$

$$\frac{\Sigma; \Gamma \vdash e_1 : \tau_1 \to \tau_2 \qquad \Sigma; \Gamma \vdash e_2 : \tau_1}{\Sigma; \Gamma \vdash e_1\ e_2 : \tau_2}\ {\to}\mathrm{E} \qquad \frac{\ell : \tau \in \Sigma}{\Sigma; \Gamma \vdash \ell : \mathsf{ref}\ \tau}\ \textsc{StoreHyp}$$

$$\frac{\Sigma; \Gamma \vdash t \div \tau}{\Sigma; \Gamma \vdash \{t\} : \mathsf{state}\ \tau}\ \textsc{state}\mathrm{I}$$

$\boxed{\Sigma; \Gamma \vdash t \div \tau}$

$$\frac{\Sigma; \Gamma \vdash e : \tau}{\Sigma; \Gamma \vdash \mathsf{ref}\ e \div \tau}\ \textsc{ref}\mathrm{I} \qquad \frac{\Sigma; \Gamma \vdash e : \mathsf{ref}\ \tau}{\Sigma; \Gamma \vdash !e \div \tau}\ \textsc{refGet}$$

$$\frac{\Sigma; \Gamma \vdash e_1 : \mathsf{ref}\ \tau \qquad \Sigma; \Gamma \vdash e_2 : \tau}{\Sigma; \Gamma \vdash e_1 := e_2 \div 1}\ \textsc{refSet} \qquad \frac{\Sigma; \Gamma \vdash e : \tau}{\Sigma; \Gamma \vdash \mathsf{return}\ e \div \tau}\ \textsc{stateRet}$$

$$\frac{\Sigma; \Gamma \vdash e : \mathsf{state}\ \tau_1 \qquad \Sigma; \Gamma, x : \tau_1 \vdash t \div \tau_2}{\Sigma; \Gamma \vdash \mathsf{let}\ x = e; t \div \tau_2}\ \textsc{stateLet}$$

$\boxed{\Sigma \vdash \sigma : \Sigma}$

$$\frac{}{\Sigma \vdash \cdot : \cdot}\ \textsc{StoreNil} \qquad \frac{\Sigma \vdash \sigma' : \Sigma' \qquad \Sigma; \cdot \vdash e : \tau}{\Sigma \vdash \sigma', \ell \mapsto v : \Sigma', \ell : \tau}\ \textsc{StoreCons}$$

$\boxed{\langle \sigma; t \rangle : \langle \Sigma; \tau \rangle}$

$$\frac{\Sigma \vdash \sigma : \Sigma \qquad \Sigma; \cdot \vdash t \div \tau}{\langle \sigma; t \rangle : \langle \Sigma; \tau \rangle}\ \textsc{CfgOk}$$

Operational Semantics

$\boxed{e \rightsquigarrow e}$

$$\frac{e \rightsquigarrow e'}{\mathbb{E}[e] \rightsquigarrow \mathbb{E}[e']}\ \textsc{EvalCtx} \qquad \frac{}{(\lambda x : \tau. e)\ v \rightsquigarrow \{v/x\}e}\ \textsc{RedFn}$$

$\boxed{\langle \sigma; e \rangle \rightsquigarrow \langle \sigma; e \rangle}$

$$\frac{e \rightsquigarrow e'}{\langle \sigma; \mathbb{T}[e] \rangle \rightsquigarrow \langle \sigma; \mathbb{T}[e'] \rangle}\ \textsc{EvalExp} \qquad \frac{\langle \sigma; t_1 \rangle \rightsquigarrow \langle \sigma'; t_1' \rangle}{\langle \sigma; \mathsf{let}\ x = \{t_1\}; t_2 \rangle \rightsquigarrow \langle \sigma'; \mathsf{let}\ x = \{t_1'\}; t_2 \rangle}\ \textsc{EvalLet}$$

$$\frac{\ell \notin \mathrm{dom}\ \sigma}{\langle \sigma; \mathsf{ref}\ v \rangle \rightsquigarrow \langle (\sigma, \ell \mapsto v); \mathsf{return}\ \ell \rangle}\ \textsc{RedRef} \qquad \frac{\sigma(\ell) = v}{\langle \sigma;\ !\ell \rangle \rightsquigarrow \langle \sigma; \mathsf{return}\ v \rangle}\ \textsc{RedGet}$$

$$\frac{}{\langle \sigma[\ell \mapsto v]; \ell := v' \rangle \rightsquigarrow \langle \sigma[\ell \mapsto v']; \mathsf{return}\ \langle \rangle \rangle}\ \textsc{RedSet}$$

$$\frac{}{\langle \sigma; \mathsf{let}\ x = \{\mathsf{return}\ v\}; t \rangle \rightsquigarrow \langle \sigma; \{v/x\}t \rangle}\ \textsc{RedLet}$$

Theorems

Lemma 3.0.1. (**Weakening, Exchange and Substitution**)

(i) **Pure** If $\Sigma; \Gamma_1, \Gamma_2 \vdash e : \tau$, then $\Sigma; \Gamma_1, x : \tau_x, \Gamma_2 \vdash e : \tau$.
Impure If $\Sigma; \Gamma_1, \Gamma_2 \vdash t \div \tau$, then $\Sigma; \Gamma_1, x : \tau_x, \Gamma_2 \vdash e \div \tau$.

(ii) **Pure** If $\Sigma; \Gamma_1, x_1 : \tau_1, x_2 : \tau_2, \Gamma_2 \vdash e : \tau$, then $\Sigma; \Gamma_1, x_2 : \tau_2, x_1 : \tau_1, \Gamma_2 \vdash e : \tau$.
Impure If $\Sigma; \Gamma_1, x_1 : \tau_1, x_2 : \tau_2, \Gamma_2 \vdash t \div \tau$, then $\Sigma; \Gamma_1, x_2 : \tau_2, x_1 : \tau_1, \Gamma_2 \vdash t \div \tau$.

(iii) **Pure** If $\Sigma; \Gamma \vdash e_1 : \tau_1$ and $\Sigma; \Gamma, x : \tau_1 \vdash e_2 : \tau_2$, then $\Sigma; \Gamma \vdash \{e_1/x\}e_2 : \tau_2$.
Impure If $\Sigma; \Gamma \vdash e : \tau_1$ and $\Sigma; \Gamma, x : \tau_1 \vdash t \div \tau_2$, then $\Sigma; \Gamma \vdash \{e_1/x\}t \div \tau_2$.

Proof. Proof by mutual *rule induction* for the **pure** and **impure** statements on $\Sigma; \Gamma_1, \Sigma_2 \vdash t \div \tau$, $\Sigma; \Gamma_1, x_1 : \tau_1, x_2 : \tau_2, \Gamma_2 \vdash t \div \tau$, and $\Sigma; \Gamma, x : \tau_1 \vdash t \div \tau_2$.

Definition 3.0.1. (**Store Extension**) We define $\Sigma_1 \leq \Sigma_2$ to mean there exists a Σ_3 s.t $\Sigma_2 = \Sigma_1, \Sigma_3$.

Lemma 3.0.2. (**Store Montonicity**) If $\Sigma_1 \leq \Sigma_2$, then:

(i) If $\Sigma_1; \Gamma \vdash e : \tau$, then $\Sigma_2; \Gamma \vdash e : \tau$.

(ii) If $\Sigma_1; \Gamma \vdash t \div \tau$, then $\Sigma_2; \Gamma \vdash t \div \tau$.

(iii) If $\Sigma_1 \vdash \sigma : \Sigma$, then $\Sigma_2 \vdash \sigma : \Sigma$.

Proof. (i) and (ii) proved by mutual *rule induction* on $\Sigma_1; \Gamma \vdash e : \tau$ and $\Sigma_1; \Gamma \vdash t \div \tau$. (iii) proved by *rule induction* on $\Sigma_1 \vdash \sigma : \Sigma$.

- Progress + Preservation for **pure** terms hold as usual.

Theorem 3.0.1. (**Progress**) If $\langle \sigma; t \rangle : \langle \Sigma; \tau \rangle$, then either $t = \mathsf{return}\ v$ or $\exists \sigma' \in \mathsf{Store}, t' \in \mathrm{I}\Lambda.\ \langle \sigma; t \rangle \rightsquigarrow \langle \sigma'; t' \rangle$.

Proof. Proof by *rule induction* on $\Sigma; \cdot \vdash t \div \tau$.

Theorem 3.0.2. (**Preservation**) If $\langle \sigma; t \rangle : \langle \Sigma; \tau \rangle$ and $\langle \sigma; t \rangle \rightsquigarrow \langle \sigma'; t' \rangle$, then there exists $\Sigma' \geq \Sigma$ s.t $\langle \sigma'; t' \rangle : \langle \Sigma'; \tau \rangle$.

Proof. Proof by *rule induction* on $\langle \sigma; t \rangle \rightsquigarrow \langle \sigma'; t' \rangle$.

- **Problem**: Monadic effects *don't* compose. e.g. cannot compose IO + State monadic effects.
- **Solutions**:
  - Adding side-effects without type-level tracking $\implies$ leads to non-termination (Landin's knot):

```
type 'a mu = ('a -> 'a) -> 'a
let fix : (int -> int) mu =
  fun f ->
    let r = ref (fun n -> 0) in
    let recur = fun n -> !r n in
    let () = r := fun n -> f recur n in
    recur
```

  - Tracking side-effects using a *type & effect system* e.g. Koka.

4 Logic

Curry-Howard Correspondence

Definition 4.0.1. (**Curry-Howard Correspondence**) The Curry-Howard correspondence defines an equivalence relation $\cong \subseteq \Lambda \times \mathsf{Prop}$:

Types	Propositions
1	$\top$
0	$\bot$
$\tau_1 + \tau_2$	$\psi \vee \phi$
$\tau_1 \times \tau_2$	$\psi \wedge \phi$
$\tau_1 \to \tau_2$	$\psi \to \phi$

If $\Delta \vdash_{\mathscr{P}} \psi$ in a proof system $\mathscr{P}$ and $\tau \cong \psi$, then there exists $e \in \Lambda$ s.t $\Gamma \vdash e : \tau$ and $\Gamma \cong \Delta$. e is the *corresponding proof* of ψ.

- Might exist *multiple* typing derivations for a given proof $\therefore$ not an *isomorphism*.
- Additional correspondences:

Logic	Programming
Propositions	Types
Proofs	Terms
Normal Forms	Values
Proof Normalization	Evaluation
Normalization Strategy	Evaluation Order

- Curry-Howard is a great way to *create* type systems.

Intuitionistic Propositional Logic

Syntax

Propositions	$P ::= \top \mid \bot \mid P \wedge P \mid P \to P \mid P \vee P$
Sequents	$\Psi ::= \cdot \mid \Psi, P$

- **Derived connectives**:

$$\neg P \triangleq P \to \bot$$

Proof System

$$\frac{P \in \Psi}{\Psi \vdash P \text{ true}} \textsc{Hyp} \qquad \frac{}{\Psi \vdash \top \text{ true}} \top\text{I} \qquad (\textsc{No elimination for } \top)$$

$$\frac{\Psi \vdash P \text{ true} \qquad \Psi \vdash Q \text{ true}}{\Psi \vdash P \land Q \text{ true}} \land\text{I} \qquad \frac{\Psi \vdash P_1 \land P_2 \text{ true}}{\Psi \vdash P_1 \text{ true}} \land\text{E}_1 \qquad \frac{\Psi \vdash P_1 \land P_2 \text{ true}}{\Psi \vdash P_2 \text{ true}} \land\text{E}_1$$

$$\frac{\Psi, P \vdash Q \text{ true}}{\Psi \vdash P \to Q \text{ true}} \to\text{I} \qquad \frac{\Psi \vdash P \to Q \text{ true} \qquad \Psi \vdash P \text{ true}}{\Psi \vdash Q \text{ true}} \to\text{E} \qquad \frac{\Psi \vdash P \text{ true}}{\Psi \vdash P \lor Q \text{ true}} \lor\text{I}_1$$

$$\frac{\Psi \vdash Q \text{ true}}{\Psi \vdash P \lor Q \text{ true}} \lor\text{I}_2 \qquad \frac{\Psi \vdash P \lor Q \text{ true} \qquad \Psi, P \vdash R \text{ true} \qquad \Psi, Q \vdash R \text{ true}}{\Psi \vdash R \text{ true}} \lor\text{E}$$

$$(\textsc{No introduction for } \bot) \qquad \frac{\Psi \vdash \bot \text{ true}}{\Psi \vdash P \text{ true}} \bot\text{E}$$

- Proofs are *constructive*
- Corresponds to simply-typed λ-calculus and *cartesian closed categories*.
- **Problem**: Cannot prove certain propositions (that are tautologies):

Name	Proposition
Law of the Excluded Middle	$\neg P \lor P$
Peirce's law	$((P \to Q) \to P) \to P$
Double Negation Elimination	$\neg\neg P \to P$
Law of Contraposition	$(\neg P \to \neg Q) \to (Q \to P)$

Classical Logic

Syntax

Propositions	$P ::= \top \mid \bot \mid P \land P \mid P \to P \mid P \lor P \mid \neg P$
True Sequents	$\Gamma ::= \cdot \mid \Gamma, P$
False Sequents	$\Delta ::= \cdot \mid \Delta, P$

Proof System

$\boxed{\Gamma; \Delta \vdash P \text{ true}}$

$$\frac{P \in \Gamma}{\Gamma; \Delta \vdash P \text{ true}} \textsc{Hyp} \qquad \frac{}{\Gamma; \Delta \vdash \top \text{ true}} \top\text{I} \qquad (\textsc{No introduction for } \bot)$$

$$\frac{\Gamma; \Delta \vdash P \ \mathsf{true} \qquad \Gamma; \Delta \vdash Q \ \mathsf{true}}{\Gamma; \Delta \vdash P \wedge Q \ \mathsf{true}} \wedge\mathrm{I} \qquad \frac{\Gamma, P; \Delta \vdash Q \ \mathsf{true}}{\Gamma; \Delta \vdash P \to Q \ \mathsf{true}} \to\mathrm{I}$$

$$\frac{\Gamma; \Delta \vdash P \ \mathsf{true}}{\Gamma; \Delta \vdash P \vee Q \ \mathsf{true}} \vee\mathrm{I}_1 \qquad \frac{\Gamma; \Delta \vdash Q \ \mathsf{true}}{\Gamma; \Delta \vdash P \vee Q \ \mathsf{true}} \vee\mathrm{I}_2 \qquad \frac{\Gamma; \Delta \vdash P \ \mathsf{false}}{\Gamma; \Delta \vdash \neg P \ \mathsf{true}} \neg\mathrm{I}$$

$$\frac{\Gamma; \Delta, P \vdash \mathsf{contr}}{\Gamma; \Delta \vdash P \ \mathsf{true}} \textsc{Contr}$$

$\boxed{\Gamma; \Delta \vdash P \ \mathsf{false}}$

$$\frac{P \in \Delta}{\Gamma; \Delta \vdash P \ \mathsf{false}} \textsc{Hyp} \qquad (\textsc{No elimination for } \top) \qquad \frac{}{\Gamma; \Delta \vdash \bot \ \mathsf{false}} \bot\mathrm{E}$$

$$\frac{\Gamma; \Delta \vdash P_1 \ \mathsf{false}}{\Gamma; \Delta \vdash P_1 \wedge P_2 \ \mathsf{false}} \wedge\mathrm{E}_1 \qquad \frac{\Gamma; \Delta \vdash P_1 \ \mathsf{false}}{\Gamma; \Delta \vdash P_1 \wedge P_2 \ \mathsf{false}} \wedge\mathrm{E}_2$$

$$\frac{\Gamma; \Delta \vdash P \ \mathsf{false} \qquad \Gamma; \Delta \vdash Q \ \mathsf{false}}{\Gamma; \Delta \vdash P \vee Q \ \mathsf{false}} \vee\mathrm{E} \qquad \frac{\Gamma; \Delta \vdash P \ \mathsf{true}}{\Gamma; \Delta \vdash \neg P \ \mathsf{false}} \neg\mathrm{E}$$

$$\frac{\Gamma; \Delta \vdash P \ \mathsf{true} \qquad \Gamma; \Delta \vdash Q \ \mathsf{false}}{\Gamma; \Delta \vdash P \to Q \ \mathsf{false}} \to\mathrm{E} \qquad \frac{\Gamma, P; \Delta \vdash \mathsf{contr}}{\Gamma; \Delta \vdash P \ \mathsf{false}} \textsc{Contr}$$

$\boxed{\Gamma; \Delta \vdash \mathsf{contr}}$

$$\frac{\Gamma; \Delta \vdash A \ \mathsf{true} \qquad \Gamma; \Delta \vdash A \ \mathsf{false}}{\Gamma; \Delta \vdash \mathsf{contr}} \textsc{Contr}$$

5 Continuations

Classical Calculus

Syntax

Types	$A ::= 1 \mid 0 \mid A \times A \mid A + A \mid A \to A \mid \neg A$
Terms	$e ::= x \mid \langle\rangle \mid \lambda x : A.e \mid \langle e, e\rangle \mid \mathsf{L}\ e \mid \mathsf{R}\ e$ $\mid \mathsf{not}\ k \mid \mu u : A.c$
Continuations	$k ::= u \mid [] \mid e :: k \mid \mathsf{fst}\ k \mid \mathsf{snd}\ k \mid [k, k]$ $\mid \mathsf{not}\ e \mid \mu x : A.c$
Contradictions	$c ::= \langle e \mid_A k\rangle$
True Contexts	$\Gamma ::= \cdot \mid \Gamma, x : A$
False Contexts	$\Delta ::= \cdot \mid \Delta, u : A$

Typing Rules

$\boxed{\Gamma; \Delta \vdash e : A\ \mathsf{true}}$

$$\frac{x : A \in \Gamma}{\Gamma; \Delta \vdash x : A\ \mathsf{true}}\ \textsc{Hyp} \qquad \frac{}{\Gamma; \Delta \vdash \langle\rangle : \top\ \mathsf{true}}\ \top\mathrm{I} \qquad (\textsc{No introduction for } \bot)$$

$$\frac{\Gamma; \Delta \vdash e_1 : A_1\ \mathsf{true} \qquad \Gamma; \Delta \vdash e_2 : A_2\ \mathsf{true}}{\Gamma; \Delta \vdash \langle e_1, e_2\rangle : A_1 \times A_2\ \mathsf{true}}\ \wedge\mathrm{I} \qquad \frac{\Gamma, x : A_1; \Delta \vdash e : A_2\ \mathsf{true}}{\Gamma; \Delta \vdash \lambda x : A_1.e : A_1 \to A_2\ \mathsf{true}}\ \to\mathrm{I}$$

$$\frac{\Gamma; \Delta \vdash e : A_1\ \mathsf{true}}{\Gamma; \Delta \vdash \mathsf{L}\ e : A_1 + A_2\ \mathsf{true}}\ \vee\mathrm{I}_1 \qquad \frac{\Gamma; \Delta \vdash e : A_2\ \mathsf{true}}{\Gamma; \Delta \vdash \mathsf{R}\ e : A_1 + A_2\ \mathsf{true}}\ \vee\mathrm{I}_2$$

$$\frac{\Gamma; \Delta \vdash k : A\ \mathsf{false}}{\Gamma; \Delta \vdash \neg\mathsf{not}\ k : A\ \mathsf{true}}\ \neg\mathrm{I} \qquad \frac{\Gamma; \Delta, u : A \vdash c\ \mathsf{contr}}{\Gamma; \Delta \vdash \mu u : A.c : A\ \mathsf{true}}\ \textsc{Contr}$$

$\boxed{\Gamma; \Delta \vdash k : A\ \mathsf{false}}$

$$\frac{u : A \in \Delta}{\Gamma; \Delta \vdash u : A\ \mathsf{false}}\ \textsc{Hyp} \qquad (\textsc{No elimination for } \top) \qquad \frac{}{\Gamma; \Delta \vdash [] : \bot\ \mathsf{false}}\ \bot\mathrm{E}$$

$$\frac{\Gamma; \Delta \vdash k : A_1 \ \mathsf{false}}{\Gamma; \Delta \vdash \mathsf{fst}\ k : A_1 \times A_2 \ \mathsf{false}} \wedge\mathrm{E}_1 \qquad \frac{\Gamma; \Delta \vdash k : A_2 \ \mathsf{false}}{\Gamma; \Delta \vdash \mathsf{snd}\ k : A_1 \times A_2 \ \mathsf{false}} \wedge\mathrm{E}_2$$

$$\frac{\Gamma; \Delta \vdash k_1 : A_1 \ \mathsf{false} \qquad \Gamma; \Delta \vdash k_2 : A_2 \ \mathsf{false}}{\Gamma; \Delta \vdash [k_1, k_2] : A_1 + A_2 \ \mathsf{false}} \vee\mathrm{E} \qquad \frac{\Gamma; \Delta \vdash e : A \ \mathsf{true}}{\Gamma; \Delta \vdash \mathsf{not}\ e : \neg A \ \mathsf{false}} \neg\mathrm{E}$$

$$\frac{\Gamma; \Delta \vdash e : A_1 \ \mathsf{true} \qquad \Gamma; \Delta \vdash k : A_2 \ \mathsf{false}}{\Gamma; \Delta \vdash e :: k : A_1 \to A_2 \ \mathsf{false}} \to\mathrm{E} \qquad \frac{\Gamma, x : A; \Delta \vdash c \ \mathsf{contr}}{\Gamma; \Delta \vdash \mu x : A.c : A \ \mathsf{false}} \textsc{Contr}$$

$\boxed{\Gamma; \Delta \vdash c \ \mathsf{contr}}$

$$\frac{\Gamma; \Delta \vdash e : A \ \mathsf{true} \qquad \Gamma; \Delta \vdash k : A \ \mathsf{false}}{\Gamma; \Delta \vdash \langle e \mid_A k \rangle \ \mathsf{contr}} \textsc{Contr}$$

Operational Semantics

- Terms and Continuations are *fully reduced* $\implies$ normal forms.

$$\frac{}{\langle \langle e_1, e_2 \rangle \mid_{A_1 \times A_2} \mathsf{fst}\ k \rangle \rightsquigarrow \langle e_1 \mid_{A_1} k \rangle} \textsc{RedFst}$$

$$\frac{}{\langle \langle e_1, e_2 \rangle \mid_{A_1 \times A_2} \mathsf{snd}\ k \rangle \rightsquigarrow \langle e_2 \mid_{A_2} k \rangle} \textsc{RedSnd}$$

$$\frac{}{\langle \mathsf{L}\ e \mid_{A_1 + A_2} [k_1, k_2] \rangle \rightsquigarrow \langle e \mid_{A_1} k_1 \rangle} \textsc{RedCase}_1 \qquad \frac{}{\langle \mathsf{R}\ e \mid_{A_1 + A_2} [k_1, k_2] \rangle} \textsc{RedCase}_2$$

$$\frac{}{\langle \lambda x : A_1.e_1 \mid_{A_1 \to A_2} e_2 :: k \rangle \rightsquigarrow \langle \{e_2/x\}e_1 \mid_{A_2} k \rangle} \textsc{RedFn}$$

$$\frac{}{\langle \mathsf{not}\ k \mid_{\neg A} \mathsf{not}\ e \rangle \rightsquigarrow \langle e \mid_A k \rangle} \textsc{RedNot}$$

$$\frac{}{\langle \mu u : A.c \mid_A k \rangle \rightsquigarrow \{k/u\}c} \textsc{RedMu}_1 \qquad \frac{}{\langle e \mid_A \mu x : A.c \rangle \rightsquigarrow \{e/x\}c} \textsc{RedMu}_2$$

- **Problem**: Non-determinism for $\langle \mu u : A.c \mid_A \mu x : A.c' \rangle$.
- **Solution**: Define an evaluation order (e.g. in STLC), since $\rightsquigarrow$ is not confluent.

Theorems

Lemma 5.0.1. (**Weakening**)
Weakening of true context Γ:

(i) If $\Gamma_1, \Gamma_2; \Delta \vdash e : B$ true, then $\Gamma_1, \Gamma_2, x : A; \Delta \vdash e : B$ true.

(ii) If $\Gamma_1, \Gamma_2; \Delta \vdash k : A$ false, then $\Gamma_1, \Gamma_2, x : A; \Delta \vdash k : A$ false.

(iii) If $\Gamma_1, \Gamma_2; \Delta \vdash c$ contr, then $\Gamma_1, \Gamma_2, x : A; \Delta \vdash c$ contr.

Weakening of false context Δ:

(i) If $\Gamma; \Delta_1, \Delta_2 \vdash e : B$ true, then $\Gamma; \Delta_1, \Delta_2, u : A \vdash e : B$ true.

(ii) If $\Gamma; \Delta_1, \Delta_2 \vdash k : B$ false, then $\Gamma; \Delta_1, \Delta_2, u : A \vdash k : B$ false.

(iii) If $\Gamma; \Delta_1, \Delta_2 \vdash c$ contr, then $\Gamma; \Delta_1, \Delta_2, u : A \vdash c$ contr.

Lemma 5.0.2. (**Exchange**)
Exchange for true context Γ:

(i) If $\Gamma_1, x_1 : A_1, x_2 : A_2, \Gamma_2; \Delta \vdash e : B$ true, then $\Gamma_1, x_2 : A_2, x_1 : A_1, \Gamma_2; \Delta \vdash e : B$ true.

(ii) If $\Gamma_1, x_1 : A_1, x_2 : A_2, \Gamma_2; \Delta \vdash k : B$ true, then $\Gamma_1, x_2 : A_2, x_1 : A_1, \Gamma_2; \Delta \vdash k : B$ true.

(iii) If $\Gamma_1, x_1 : A_1, x_2 : A_2, \Gamma_2; \Delta \vdash c$ contr, then $\Gamma_1, x_2 : A_2, x_1 : A_1, \Gamma_2; \Delta \vdash c$ contr.

Exchange for false context Δ:

- If $\Gamma; \Delta_1, x_1 : A_1, x_2 : A_2, \Delta_2 \vdash e : B$ true, then $\Gamma; \Delta_1, x_2 : A_2, x_1 : A_1; \Delta_2 \vdash e : B$ true.
- If $\Gamma; \Delta_1, x_1 : A_1, x_2 : A_2, \Delta_2 \vdash k : B$ false, then $\Gamma; \Delta_1, x_2 : A_2, x_1 : A_1; \Delta_2 \vdash k : B$ false.
- If $\Gamma; \Delta_1, x_1 : A_1, x_2 : A_2, \Delta_2 \vdash c$ contr, then $\Gamma; \Delta_1, x_2 : A_2, x_1 : A_1; \Delta_2 \vdash c$ contr.

Lemma 5.0.3. (**Substitution**)
If $\Gamma; \Delta \vdash e : A$ true, then

(i) If $\Gamma, x : A; \Delta \vdash e' : B$ true, then $\Gamma; \Delta \vdash \{e/x\}e' : B$ true.

(ii) If $\Gamma, x : A; \Delta \vdash k : B$ false, then $\Gamma; \Delta \vdash \{e/x\}k : B$ false.

(iii) If $\Gamma, x : A; \Delta \vdash c$ contr, then $\Gamma; \Delta \vdash \{e/x\}c$ contr.

If $\Gamma; \Delta \vdash k : A$ false, then

(i) If $\Gamma; \Delta, u : A \vdash e : B$ true, then $\Gamma; \Delta \vdash \{k/u\}e : B$ true.

(ii) If $\Gamma; \Delta, u : A \vdash k' : B$ false, then $\Gamma; \Delta \vdash \{k/u\}k' : B$ false.

(iii) If $\Gamma; \Delta, u : A \vdash c$ contr, then $\Gamma; \Delta \vdash \{k/u\}c$ contr.

Theorem 5.0.1. (**Preservation**) If $\cdot; \cdot \vdash c$ contr and $c \rightsquigarrow c'$, then $\cdot; \cdot \vdash c'$ contr.

Proof. Proof by *case analysis* on $c \rightsquigarrow c'$.

- **Problem**: $\cdot; \cdot \vdash c$ contr is a contradiction $\therefore$ (assuming consistency) c cannot exist.
- **Solution**: Add a normal form for closed contradictions.

(ans: runtime type, halt: terminating instruction, done: empty stack)

Types	$A ::= \ldots \mid \mathsf{ans}$
Terms	$e ::= \ldots \mid \mathsf{halt}$
Continuations	$k ::= \ldots \mid \mathsf{done}$

$$\frac{}{\Gamma; \Delta \vdash \mathsf{halt} : \mathsf{ans\ true}}\ \textsc{AnsI} \qquad \frac{}{\Gamma; \Delta \vdash \mathsf{done} : \mathsf{ans\ false}}\ \textsc{AnsE}$$

Theorem 5.0.2. (**Progress**) If $\cdot; \cdot \vdash c$ contr, then $\exists c' \in \mathsf{Contr}.c \rightsquigarrow c'$ or $c = \langle \mathsf{halt} \mid_{\mathsf{ans}} \mathsf{done} \rangle$.

Proof. Proof by *rule induction* on $\cdot; \cdot \vdash c$ contr.

Continuation Translation

- **Idea**: Translate classical calculus (classical logic) to STLC (intuitionistic Propositional logic)
- Let $\mathsf{ans} \in \mathsf{Type}$ be a STLC type (*intuitively*, it is the runtime type).

Definition 5.0.1. (**Quasi-negation Translations**) The *quasi-negation* of the type τ is $\sim \tau \triangleq \tau \to \mathsf{ans}$. The quasi-negation translations for types A, true contexts Γ, and false contexts Δ:

$$\begin{aligned}
(\neg A)^\circ &= {\sim} A^\circ \\
(1)^\circ &= 1 \\
(0)^\circ &= \mathsf{ans} \\
(A \times B)^\circ &= A^\circ \times B^\circ \\
(A + B)^\circ &= {\sim\sim}(A^\circ + B^\circ) \\
(A \to B)^\circ &= {\sim\sim}(A^\circ \to B^\circ) \\
\\
(\cdot)^\circ &= \cdot \\
(\Gamma, x : A)^\circ &= \Gamma^\circ, x : A^\circ \\
\\
{\sim}(\cdot) &= \cdot \\
{\sim}(\Delta, u : A) &= {\sim}\Delta, x : {\sim} A^\circ
\end{aligned}$$

Definition 5.0.2. (**Double and Triple Negation Elimination**) For all $\tau \in \mathsf{Type}$, triple-negation eliminator $\cdot \vdash \mathsf{tne}_\tau : {\sim\sim\sim}\tau \to {\sim}\tau$

$$\mathsf{tne}_\tau \triangleq \lambda k : {\sim\sim\sim}\tau.\lambda x : \tau.k\ (\lambda q : {\sim\sim}\tau.q\ x)$$

For all $A \in \mathsf{Type}$, double-negation eliminator is defined inductively $\cdot \vdash \mathsf{dne}_A : {\sim\sim} A^\circ \to A^\circ$:

$$\begin{aligned}
\mathsf{dne}_1 &= \lambda q : {\sim\sim}1.\,\langle\rangle \\
\mathsf{dne}_0 &= \lambda q : {\sim\sim}0.q\ (\lambda x : \mathsf{ans}.x) \\
\mathsf{dne}_{A\times B} &= \lambda q : {\sim\sim}(A^\circ \times B^\circ).\Big\langle \mathsf{dne}_A\ (\lambda k : {\sim}A^\circ.q\ (\lambda p : A^\circ \times B^\circ.k\ (\mathsf{fst}\ p))), \\
&\qquad \mathsf{dne}_B\ (\lambda k : {\sim}B^\circ.q\ (\lambda p : A^\circ \times B^\circ.k\ (\mathsf{snd}\ p)))\Big\rangle \\
\mathsf{dne}_{A+B} &= \lambda q : {\sim\sim\sim\sim}(A^\circ + B^\circ).\mathsf{tne}_{A^\circ + B^\circ}\ q \\
\mathsf{dne}_{\neg A} &= \lambda q : {\sim\sim\sim}A^\circ.\mathsf{tne}_{A^\circ}\ q \\
\mathsf{dne}_{A\to B} &= \lambda q : {\sim\sim\sim\sim}(A^\circ \to B^\circ).\mathsf{tne}_{A^\circ \to B^\circ}\ q
\end{aligned}$$

Definition 5.0.3. (**Classical Embedding**) The classical embedding of the terms e, continuations k and contradictions c are given by:

$$\langle e \mid_A k\rangle^\circ = k^\circ\ e^\circ$$

$$\begin{aligned}
x^\circ &= x \\
\langle\rangle^\circ &= \langle\rangle
\end{aligned}$$

$$
\begin{aligned}
\langle e_1, e_2\rangle^\circ &= \langle e_1^\circ, e_2^\circ\rangle \\
(\mathsf{L}\ e)^\circ &= \lambda k : {\sim}(A^\circ + B^\circ).k\ (\mathsf{L}\ e^\circ) \\
(\mathsf{R}\ e)^\circ &= \lambda k : {\sim}(A^\circ + B^\circ).k\ (\mathsf{R}\ e^\circ) \\
(\mathsf{not}\ k)^\circ &= k^\circ \\
(\lambda x : A.e)^\circ &= \lambda k : {\sim}(A^\circ \to B^\circ).k\ (\lambda x : A^\circ.e^\circ) \\
(\mu u : A.c)^\circ &= \mathsf{dne}_A\ (\lambda u : {\sim}A^\circ.c^\circ)
\end{aligned}
$$

$$
\begin{aligned}
u^\circ &= u \\
[]^\circ &= \lambda x : \mathsf{ans}.x \\
[k_1, k_2]^\circ &= \lambda k : {\sim}{\sim}(A^\circ + B^\circ).k\ (\lambda i : A^\circ + B^\circ. \\
&\qquad\qquad \mathsf{case}(i, \mathsf{L}\ x_1 \to k_1^\circ\ x_1, \mathsf{R}\ x_2 \to k_2^\circ\ x_2)) \\
(\mathsf{fst}\ k)^\circ &= \lambda q : (A^\circ \times B^\circ).k^\circ\ (\mathsf{fst}\ q) \\
(\mathsf{snd}\ k)^\circ &= \lambda q : (A^\circ \times B^\circ).k^\circ\ (\mathsf{snd}\ q) \\
(\mathsf{not}\ e)^\circ &= \lambda k : {\sim}A^\circ.k\ e^\circ \\
(e :: k)^\circ &= \lambda q : {\sim}{\sim}(A^\circ \to B^\circ).q\ (\lambda p : A^\circ \to B^\circ.k^\circ\ (p\ e^\circ)) \\
(\mu x : A.c)^\circ &= \lambda x : A^\circ.c^\circ
\end{aligned}
$$

Theorem 5.0.3. Classical terms encode the corresponding types in STLC:

(i) If $\Gamma; \Delta \vdash e : A$ true, then $\Gamma^\circ,\ {\sim}\Delta \vdash e^\circ : A^\circ$.

(ii) If $\Gamma; \Delta \vdash k : A$ false, then $\Gamma^\circ,\ {\sim}\Delta \vdash k^\circ : {\sim}A^\circ$.

(iii) If $\Gamma; \Delta \vdash c$ contr, then $\Gamma^\circ,\ {\sim}\Delta \vdash c^\circ :$ ans.

Proof. Proof by *rule induction* on derivations.

Continuation Calculus

Syntax

Types	$A ::= 1 \mid 0 \mid A \times A \mid A + A \mid A \to A \mid \neg A$
Terms	$e ::= x \mid \langle\rangle \mid \lambda x : A.e \mid e\ e \mid \langle e, e\rangle \mid \mathsf{fst}\ e \mid \mathsf{snd}\ e$ $\mid \mathsf{L}\ e \mid \mathsf{R}\ e \mid \mathsf{case}(e, \mathsf{L}\ x \to e, \mathsf{R}\ x \to e)$ $\mid \mathsf{abort}\ e \mid \mathsf{cont}\ x\ \mathsf{in}\ e \mid \mathsf{throw}_A(e, e)$
Contexts	$\Gamma ::= \cdot \mid \Gamma, x : A$

Typing Rules

$$\frac{x : A \in \Gamma}{\Gamma \vdash x : A}\ \textsc{Hyp} \qquad \frac{}{\Gamma \vdash \langle\rangle : 1}\ 1\mathrm{I} \qquad (\textsc{No elimination for } 1)$$

(No introduction for 0) $$\frac{\Gamma \vdash e : 0}{\Gamma \vdash \mathsf{abort}\ e : A}\ 0\mathrm{E}$$

$$\frac{\Gamma \vdash e_1 : A_1 \qquad \Gamma \vdash e_2 : A_2}{\Gamma \vdash \langle e_1, e_2 \rangle : A_1 \times A_2}\ \times\mathrm{I} \qquad \frac{\Gamma \vdash e : A_1 \times A_2}{\Gamma \vdash \mathsf{fst}\ e : A_1}\ \times\mathrm{E}_1 \qquad \frac{\Gamma \vdash e : A_1 \times A_2}{\Gamma \vdash \mathsf{snd}\ e : A_2}\ \times\mathrm{E}_2$$

$$\frac{\Gamma, x : A_1 \vdash e : A_2}{\Gamma \vdash \lambda x : A_1 . e : A_1 \to A_2}\ \to\mathrm{I} \qquad \frac{\Gamma \vdash e_1 : A_1 \to A_2 \qquad \Gamma \vdash e_2 : A_1}{\Gamma \vdash e_1\ e_2 : A_2}\ \to\mathrm{E}$$

$$\frac{\Gamma \vdash e : A_1}{\Gamma \vdash \mathsf{L}\ e : A_1 + A_2}\ +\mathrm{I}_1 \qquad \frac{\Gamma \vdash e : A_2}{\Gamma \vdash \mathsf{R}\ e : A_1 + A_2}\ +\mathrm{I}_2$$

$$\frac{\Gamma \vdash e : A_1 + A_2 \qquad \Gamma, x_1 : A_1 \vdash e_1 : A \qquad \Gamma, x_2 : A_2 \vdash e_2 : A}{\Gamma \vdash \mathsf{case}(e, \mathsf{L}\ x_1 \to e_1, \mathsf{R}\ x_2 \to e_2) : A}\ +\mathrm{E}$$

$$\frac{\Gamma, x : \neg A \vdash e : A}{\Gamma \vdash \mathsf{cont}\ x\ \mathsf{in}\ e : A}\ \neg\mathrm{I} \qquad \frac{\Gamma \vdash e_1 : \neg A \qquad \Gamma \vdash e_2 : A}{\Gamma \vdash \mathsf{throw}_B(e_1, e_2) : B}\ \neg\mathrm{E}$$

Continuation Translation

Definition 5.0.4. (**CPS Translation**) The CPS translation of the type A, contexts Γ and terms e:

$$\begin{aligned}
\neg A^\bullet &= \sim\sim\sim A^\bullet \\
1^\bullet &= \sim\sim 1 \\
0^\bullet &= \sim\sim 0 \\
(A \to B)^\bullet &= \sim\sim(A^\bullet \to B^\bullet) \\
(A \times B)^\bullet &= \sim\sim(A^\bullet \times B^\bullet) \\
(A + B)^\bullet &= \sim\sim(A^\bullet + B^\bullet)
\end{aligned}$$

$$\begin{aligned}
(\cdot)^\bullet &= \cdot \\
(\Gamma, x : A)^\bullet &= \Gamma^\bullet, x : A^\bullet
\end{aligned}$$

$$\begin{aligned}
x^\bullet &= \lambda k : \sim A^\bullet . k\ x \\
\langle\rangle^\bullet &= \lambda k : \sim 1 . k\ \ \langle\rangle \\
\langle e_1, e_2 \rangle^\bullet &= \lambda k : \sim(A^\bullet \times B^\bullet) . e_1^\bullet\ (\lambda x : A^\bullet . e_2^\bullet\ (\lambda y : B^\bullet . k\ \ \langle x, y \rangle)) \\
(\mathsf{fst}\ e)^\bullet &= \lambda k : \sim A^\bullet . e^\bullet\ (\lambda p : A^\bullet \times B^\bullet . k\ (\mathsf{fst}\ p)) \\
(\mathsf{snd}\ e)^\bullet &= \lambda k : \sim B^\bullet . e^\bullet\ (\lambda p : A^\bullet \times B^\bullet . k\ (\mathsf{snd}\ p)) \\
(\mathsf{L}\ e)^\bullet &= \lambda k : \sim(A^\bullet + B^\bullet) . e^\bullet\ (\lambda x : A^\bullet . k\ (\mathsf{L}\ x)) \\
(\mathsf{R}\ e)^\bullet &= \lambda k : \sim(A^\bullet + B^\bullet) . e^\bullet\ (\lambda x : B^\bullet . k\ (\mathsf{R}\ x)) \\
\mathsf{case}(e, \mathsf{L}\ x_1 \to e_2, \mathsf{R}\ x_2 \to e_2)^\bullet &= \lambda k : \sim C^\bullet . e^\bullet\ (\lambda v : A^\bullet + B^\bullet . \\
&\qquad \mathsf{case}(v, \mathsf{L}\ x_1 \to e_1^\bullet\ k, \mathsf{R}\ x_2 \to e_2^\bullet\ k)) \\
(\lambda x : A . e)^\bullet &= \lambda k : \sim(A^\bullet \to B^\bullet) . k\ (\lambda x : A^\bullet . e^\bullet)
\end{aligned}$$

$$
\begin{aligned}
(e_1\ e_2)^\bullet &= \lambda k : {\sim}B^\bullet.e_1^\bullet\ (\lambda f : A^\bullet \to B^\bullet.e_2^\bullet\ (\lambda x : A^\bullet.k\ (f\ x))) \\
(\mathsf{abort}\ e)^\bullet &= \lambda k : {\sim}A.e^\bullet\ (\lambda x : 0.k\ (\mathsf{abort}\ x)) \\
(\mathsf{cont}\ x\ \mathsf{in}\ e)^\bullet &= \lambda k : {\sim}A^\bullet.\mathsf{let}\ x = \lambda q : {\sim}{\sim}A^\bullet.q\ k\ \mathsf{in}\ e^\bullet\ k \\
(\mathsf{throw}_B(e_1, e_2))^\bullet &= \lambda k : {\sim}B^\bullet.(\mathsf{tne}_A\ e_1^\bullet)\ e_2^\bullet
\end{aligned}
$$

Theorem 5.0.4. If $\Gamma \vdash e : A$, then $\Gamma^\bullet \vdash e^\bullet : A^\bullet$.

Proof. Proof by *rule induction* on $\Gamma \vdash e : A$.

6 Dependent Types

Syntax

Types, Terms $\quad A, e ::= x$

$\quad\quad\quad \mid 1 \mid \langle\rangle$

$\quad\quad\quad \mid \Pi x : A.B \mid \lambda x : A.e \mid e\ e$

$\quad\quad\quad \mid (e = e : A) \mid \mathsf{refl}\ e \mid \mathsf{subst}[x : A.B](e, e)$

Contexts $\quad \Gamma ::= \cdot \mid \Gamma, x : A$

Typing Rules / Operational Semantics

(F: Formation, Red: Reduction, Cong: Congruence)

$\boxed{\Gamma\ \mathsf{ctx}}$

$$\frac{}{\cdot\ \mathsf{ctx}} \qquad \frac{\Gamma\ \mathsf{ctx} \qquad \Gamma \vdash A\ \mathsf{type}}{\Gamma, x : A\ \mathsf{ctx}}$$

$\boxed{\Gamma \vdash A\ \mathsf{type}}$

$$\frac{}{\Gamma \vdash 1\ \mathsf{type}}\ 1\mathrm{F} \qquad \frac{\Gamma \vdash A\ \mathsf{type} \qquad \Gamma, x : A \vdash B\ \mathsf{type}}{\Gamma \vdash \Pi x : A.B\ \mathsf{type}}\ \Pi\mathrm{F}$$

$$\frac{\Gamma \vdash A\ \mathsf{type} \qquad \Gamma \vdash e_1 : A \qquad \Gamma \vdash e_2 : A}{\Gamma \vdash (e_1 = e_2 : A)\ \mathsf{type}}\ \mathrm{eqF}$$

$\boxed{\Gamma \vdash e : A}$

$$\frac{x : A \in \Gamma}{\Gamma \vdash x : A}\ \mathrm{Hyp} \qquad \frac{}{\Gamma \vdash \langle\rangle : 1}\ 1\mathrm{I} \qquad \text{(No elimination for 1)}$$

$$\frac{\Gamma \vdash A\ \mathsf{type} \qquad \Gamma, x : A \vdash e : B}{\Gamma \vdash \lambda x : A.e : \Pi x : A.B}\ \Pi\mathrm{I} \qquad \frac{\Gamma \vdash e_1 : \Pi x : A.B \qquad \Gamma \vdash e_2 : A}{\Gamma \vdash e_1\ e_2 : \{e_2/x\}B}\ \Pi\mathrm{E}$$

$$\frac{\Gamma \vdash e : A \qquad \Gamma \vdash A \equiv B\ \mathsf{type}}{\Gamma \vdash e : B}\ \mathrm{Cong} \qquad \frac{\Gamma \vdash e : A}{\Gamma \vdash \mathsf{refl}\ e : (e = e : A)}\ \mathrm{eqI}$$

$$\frac{\Gamma \vdash A\ \mathsf{type} \qquad \Gamma, x : A \vdash B\ \mathsf{type} \qquad \Gamma \vdash e_{\mathrm{eq}} : (e_1 = e_2 : A) \qquad \Gamma \vdash e_P : \{e_1/x\}B}{\Gamma \vdash \mathsf{subst}[x : A.B](e_{\mathrm{eq}}, e_P) : \{e_2/x\}B}\ \mathrm{eqE}$$

$\boxed{\Gamma \vdash A \equiv B \ \mathsf{type}}$

$$\frac{}{\Gamma \vdash 1 \equiv 1 \ \mathsf{type}}\ \textsc{Cong1} \qquad \frac{\Gamma \vdash A \equiv X \ \mathsf{type} \qquad \Gamma, x : A \vdash B \equiv Y \ \mathsf{type}}{\Gamma \vdash \Pi x : A.B \equiv \Pi x : X.Y \ \mathsf{type}}\ \textsc{Cong}\Pi$$

$$\frac{\Gamma \vdash e_{1i} : A_1 \qquad \Gamma \vdash e_{2i} : A_2 \qquad \Gamma \vdash A_1 \equiv A_2 \ \mathsf{type} \qquad \Gamma \vdash e_{1i} \equiv e_{2i} : A}{\Gamma \vdash (e_{11} = e_{12} : A_1) \equiv (e_{21} = e_{22} : A_2) \ \mathsf{type}}\ \textsc{CongEq}$$

$\boxed{\Gamma \vdash e \equiv e : A}$

$$\frac{\Gamma \vdash e : A}{\Gamma \vdash e \equiv e : A}\ \textsc{EquivRefl} \qquad \frac{\Gamma \vdash e_2 \equiv e_1 : A}{\Gamma \vdash e_1 \equiv e_2 : A}\ \textsc{EquivSym}$$

$$\frac{\Gamma \vdash e_1 \equiv e_2 : A \qquad \Gamma \vdash e_2 \equiv e_3 : A}{\Gamma \vdash e_1 \equiv e_3 : A}\ \textsc{EquivTrans} \qquad \frac{x : A \in \Gamma}{\Gamma \vdash x \equiv x : A}\ \textsc{CongHyp}$$

$$\frac{}{\Gamma \vdash \langle\rangle \equiv \langle\rangle : 1}\ \textsc{Cong1I} \qquad \frac{\Gamma \vdash A_1 \equiv A_2 \ \mathsf{type} \qquad \Gamma, x : A_1 \vdash e_1 \equiv e_2 : B}{\Gamma \vdash \lambda x : A_1.e_1 \equiv \lambda x : A_2.e_2 : \Pi x : A_1.B}\ \textsc{Cong}\Pi\textsc{I}$$

$$\frac{\Gamma \vdash e_{11} = e_{21} : \Pi x : A.B \qquad \Gamma \vdash e_{12} \equiv e_{22} : A}{\Gamma \vdash e_{11}\ e_{12} \equiv e_{21}\ e_{22} : \{e_{12}/x\}B}\ \textsc{Cong}\Pi\textsc{E}$$

$$\frac{\Gamma \vdash e_1 \equiv e_2 : A \qquad \Gamma \vdash A \equiv B \ \mathsf{type}}{\Gamma \vdash e_1 \equiv e_2 : B}\ \textsc{Cong}$$

$$\frac{\Gamma \vdash e_1 \equiv e_2 : A}{\Gamma \vdash \mathsf{refl}\ e_1 \equiv \mathsf{refl}\ e_2 : (e_1 = e_2 : A)}\ \textsc{CongEqI}$$

$$\frac{\begin{array}{c}\Gamma \vdash A_1 \equiv A_2 \ \mathsf{type} \qquad \Gamma, x : A_1 \vdash B_1 \equiv B_2 \ \mathsf{type} \\ \Gamma \vdash e_{\mathsf{eq}1} \equiv e_{\mathsf{eq}2} : (e_1 = e_2 : A_1) \qquad \Gamma \vdash e_{P1} \equiv e_{P2} : \{e_1/x\}B_1\end{array}}{\Gamma \vdash \mathsf{subst}[x : A_1.B_1](e_{\mathsf{eq}1}, e_{P1}) \equiv \mathsf{subst}[x : A_2.B_2](e_{\mathsf{eq}2}, e_{P2}) : \{e_2/x\}B_1}\ \textsc{CongEqE}$$

$$\frac{\Gamma \vdash \lambda x : A.e : \Pi x : A.B \qquad \Gamma \vdash e_2 : A \qquad \Gamma \vdash \{e_2/x\}e_1 : \{e_2/x\}B}{\Gamma \vdash (\lambda x : A.e_1)\ e_2 \equiv \{e_2/x\}e_1 : \{e_2/x\}B}\ \textsc{Red}\Pi$$

$$\frac{\Gamma \vdash \mathsf{subst}[x : A.B](\mathsf{refl}\ e, e_P) : \{e/x\}B \qquad \Gamma \vdash e_P : \{e/x\}B}{\Gamma \vdash \mathsf{subst}[x : A.B](\mathsf{refl}\ e, e_P) \equiv e : \{e/x\}B}\ \textsc{RedEq}_1$$

Theorems

Lemma 6.0.1. (**Weakening**) If $\Gamma_1 \vdash X$ type, then

(i) If $\Gamma_1, \Gamma_2 \vdash A$ type, then $\Gamma_1, x : X, \Gamma_2 \vdash A$ type.

(ii) If $\Gamma_1, \Gamma_2 \vdash e : A$, then $\Gamma_1, x : X, \Gamma_2 \vdash e : A$.

(iii) If $\Gamma_1, \Gamma_2 \vdash A \equiv B$ type, then $\Gamma_1, x : X, \Gamma_2 \vdash A \equiv B$ type.

(iv) If $\Gamma_1, \Gamma_2 \vdash e_1 \equiv e_2 : A$, then $\Gamma_1, x : X, \Gamma_2 \vdash e_1 \equiv e_2 : A$.

(v) If Γ_1, Γ_2 ctx, then $\Gamma_1, x : X, \Gamma_2$ ctx.

Proof. Proof by mutual *rule induction* on (i)-(iv). Proof by *rule induction* on (v).

Lemma 6.0.2. (**Substitution**) If $\Gamma_1 \vdash e_x : X$, then

(i) If $\Gamma_1, x : X, \Gamma_2 \vdash A$ type, then $\Gamma_1, \{e_x/x\}\Gamma_2 \vdash \{e_x/x\}A$ type.

(ii) If $\Gamma_1, x : X, \Gamma_2 \vdash e : A$, then $\Gamma_1, \{e_x/x\}\Gamma_2 \vdash \{e_x/x\}e : \{e_x/x\}A$.

(iii) If $\Gamma_1, x : X, \Gamma_2 \vdash A \equiv B$ type, then $\Gamma_1, \{e_x/x\}\Gamma_2 \vdash \{e_x/x\}A \equiv \{e_x/x\}B$ type.

(iv) If $\Gamma_1, x : X, \Gamma_2 \vdash e_1 \equiv e_2 : A$, then $\Gamma_1, \{e_x/x\}\Gamma_2 \vdash \{e_x/x\}e_1 \equiv \{e_x/x\}e_2 : \{e_x/x\}A$.

(v) If $\Gamma_1, x : X, \Gamma_2$ ctx, then $\Gamma_1, \{e_x/x\}\Gamma_2$ ctx.

Proof. Proof by mutual *rule induction* on (i)-(iv). Proof by *rule induction* on (v).

Lemma 6.0.3. (**Context Equality**) If $\Gamma_1 \vdash X \equiv Y$ type, then

(i) If $\Gamma_1, x : X, \Gamma_2 \vdash A$ type, tehen $\Gamma_1, x : Y, \Gamma_2 \vdash A$ type.

(ii) If $\Gamma_1, x : X, \Gamma_2 \vdash e : A$, then $\Gamma_1, x : Y, \Gamma_2 \vdash e : A$.

(iii) If $\Gamma_1, x : X, \Gamma_2 \vdash A \equiv B$ type, then $\Gamma_1, x : Y, \Gamma_2 \vdash A \equiv B$ type.

(iv) If $\Gamma_1, x : X, \Gamma_2 \vdash e_1 \equiv e_2 : A$, then $\Gamma_1, x : Y, \Gamma_2 \vdash e_1 \equiv e_2 : A$.

(v) If $\Gamma_1, x : X, \Gamma_2$ ctx, then $\Gamma_1, x : Y, \Gamma_2$ ctx.

Proof. Proof by mutual *rule induction* on (i)-(iv). Proof by *rule induction* on (v).

Theorem 6.0.1. (**Regularity**) If Γ ctx, then

(i) If $\Gamma \vdash e : A$, then $\Gamma \vdash A$ type.

(ii) If $\Gamma \vdash A \equiv B$ type, then $\Gamma \vdash A$ type and $\Gamma \vdash B$ type.