Typesetting λ-calculi with LaTeX

Marco Patrignani

This document contains a reference of the syntax and semantics for ULC and STLC.

Syntax highlighting is nice way to guide you in visually separating each language; please inform me whether they cause any distraught.

Please read the latex comments too.

Contents

1 λ^u: Untyped Lambda Calculus

1.1 Syntax

$$\begin{aligned} t &::= n \mid \lambda x.\, t \mid x \mid t\ t \mid t \oplus t \\ v &::= n \mid \lambda x.\, t \\ \Omega &::= t \mid \mathit{fail} \\ \mathbb{E} &::= [\cdot] \mid \mathbb{E}\ t \mid (\lambda x.\, t)\ \mathbb{E} \mid \mathbb{E} \oplus t \mid n \oplus \mathbb{E} \end{aligned}$$

$\oplus$ ranges over $+, -, *$.

1.2 Structural Operational Semantics

1.2.1 Small Step, Call by Value (SSV)

$$\Omega \hookrightarrow \Omega \qquad \text{judgement}$$

Reductions.

$$\frac{}{(\lambda x.\, t)\ v \hookrightarrow t[v\ /\ x]}\ \text{(SSV-Beta)} \qquad \frac{}{n \oplus n' \hookrightarrow n[\![\oplus]\!]n'}\ \text{(SSV-Op)} \qquad \frac{t_1 \hookrightarrow t_1'}{t_1\ t_2 \hookrightarrow t_1'\, t_2}\ \text{(SSV-App1)}$$

$$\frac{t_2 \hookrightarrow t_2'}{(\lambda x.\, t)\ t_2 \hookrightarrow (\lambda x.\, t)\ t_2'}\ \text{(SSV-App2)} \qquad \frac{t_1 \hookrightarrow t_1'}{t_1 \oplus t_2 \hookrightarrow t_1' \oplus t_2}\ \text{(SSV-Op1)} \qquad \frac{t_2 \hookrightarrow t_2'}{n \oplus t_2 \hookrightarrow n \oplus t_2'}\ \text{(SSV-Op2)}$$

Fail reductions.

$$\frac{}{(\lambda x.\, t) \oplus t \hookrightarrow \mathit{fail}}\ \text{(SSV-Op-fail-l)} \qquad \frac{}{n\ \oplus (\lambda x.\, t) \hookrightarrow \mathit{fail}}\ \text{(SSV-Op-fail-r)} \qquad \frac{}{n\ t \hookrightarrow \mathit{fail}}\ \text{(SSV-App-fail-fun)}$$

$$\frac{t_2 \hookrightarrow \mathit{fail}}{(\lambda x.\, t)\ t_2 \hookrightarrow \mathit{fail}}\ \text{(SSV-App-fail-arg)} \qquad \frac{t_1 \hookrightarrow \mathit{fail}}{t_1\ t_2 \hookrightarrow \mathit{fail}}\ \text{(SSV-App-fail-1)} \qquad \frac{t_1 \hookrightarrow \mathit{fail}}{t_1 \oplus t_2 \hookrightarrow \mathit{fail}}\ \text{(SSV-Op-fail-1)}$$

$$\frac{t_2 \hookrightarrow \mathit{fail}}{n \oplus t_2 \hookrightarrow \mathit{fail}}\ \text{(SSV-Op-fail-2)}$$

1.2.2 Small Step, Call by Name (SSN)

Remove Rule SSV-App2 and replace Rule SSV-Beta with:

$$\frac{}{(\lambda x.\, t)\ t' \hookrightarrow t[t'\ /\ x]}\ \text{(SSN-Beta)}$$

1.2.3 Big Step, Call by Value (SBV)

$$\mathit{\Omega} \Downarrow \mathit{\Omega} \qquad\qquad \text{judgement}$$

Reductions. Reductions for failing are omitted.

$$\frac{}{v \Downarrow v}\ \text{(SBV-val)} \qquad \frac{t \Downarrow n \qquad t' \Downarrow n'}{t \oplus t' \Downarrow n[\![\oplus]\!]n'}\ \text{(SBV-op)} \qquad \frac{t \Downarrow \lambda x.\, t'' \qquad t' \Downarrow v \qquad t''[v \;/\; x] \Downarrow v'}{t\ t' \Downarrow v'}\ \text{(SBV-app)}$$

1.3 Contextual Operational Semantics, Call by Value (CSV)

$$\mathit{\Omega} \rightsquigarrow \mathit{\Omega} \qquad\qquad \text{judgement}$$
$$\mathit{\Omega} \rightsquigarrow^{\mathrm{p}} \mathit{\Omega} \qquad\qquad \text{judgement}$$

Reductions. Reductions for failing are omitted.

$$\frac{t \rightsquigarrow^{\mathrm{p}} t'}{\mathbb{E}\,[t] \rightsquigarrow \mathbb{E}\,[t']}\ \text{(CSV-ctx)}$$

Plus Rules SSV-Beta and SSV-Op, changing $\hookrightarrow$ with $\rightsquigarrow^{\mathrm{p}}$.

2 λ^{τ}: Simply-Typed Lambda Calculus

2.1 Syntax

No changes save for program state.

$$\begin{aligned} \Omega &::= \mathsf{t} \\ \Gamma &::= \emptyset \mid \Gamma, (\mathsf{x} : \tau) \\ \tau &::= \mathsf{N} \mid \tau \to \tau \end{aligned}$$

2.2 Static Semantics

$$\Gamma \vdash \mathsf{t} : \tau \qquad\qquad \text{judgement}$$

Reductions.

$$\frac{\mathsf{x} : \tau \in \Gamma}{\Gamma \vdash \mathsf{x} : \tau}\ \text{(Type-var)} \qquad \frac{\Gamma, \mathsf{x} : \tau \vdash \mathsf{t} : \tau'}{\Gamma \vdash \lambda \mathsf{x} : \tau.\, \mathsf{t} : \tau \to \tau'}\ \text{(Type-lam)} \qquad \frac{}{\Gamma \vdash \mathsf{n} : \mathsf{N}}\ \text{(Type-num)}$$

$$\frac{\Gamma \vdash \mathsf{t} : \tau' \to \tau \qquad \Gamma \vdash \mathsf{t}' : \tau'}{\Gamma \vdash \mathsf{t}\ \mathsf{t}' : \tau}\ \text{(Type-app)} \qquad \frac{\Gamma \vdash \mathsf{t} : \mathsf{N} \qquad \Gamma \vdash \mathsf{t}' : \mathsf{N}}{\Gamma \vdash \mathsf{t}\ \mathsf{t}' : \mathsf{N}}\ \text{(Type-op)}$$

2.3 Contextual Operational Semantics, Call by Value (CSV)

No changes, repeated for clarity.

$$\Omega \rightsquigarrow \Omega \qquad \text{judgement}$$
$$\Omega \rightsquigarrow^{\mathrm{p}} \Omega \qquad \text{judgement}$$

Reductions. No reductions for failing.

$$\frac{t \rightsquigarrow^{\mathrm{p}} t'}{\mathbb{E}\,[t] \rightsquigarrow \mathbb{E}\,[t']}\ \text{(CSV-ctx)} \qquad \frac{}{(\lambda x.\, t)\ v \rightsquigarrow^{\mathrm{p}} t[v\ /\ x]}\ \text{(CSV-Beta)} \qquad \frac{}{n \oplus n' \rightsquigarrow^{\mathrm{p}} n [\![\oplus]\!] n'}\ \text{(CSV-Op)}$$