PCF

PCF

Terms and types

Types: $\tau ::= \texttt{nat} \mid \texttt{bool} \mid \tau \to \tau$

Syntax of PCF

Types:

$$\tau ::= \texttt{nat} \mid \texttt{bool} \mid \tau \to \tau$$

Terms:

$$\begin{aligned} t \quad ::= \quad & \texttt{0} \mid \texttt{succ}(t) \mid \texttt{pred}(t) \mid \\ & \texttt{true} \mid \texttt{false} \mid \texttt{zero?}(t) \mid \texttt{if}\ t\ \texttt{then}\ t\ \texttt{else}\ t \\ & x \mid \texttt{fun}\ x{:}\tau.\ t \mid t\ t \mid \texttt{fix}(t) \end{aligned}$$

Typing for PCF (I)

$\boxed{\Gamma \vdash t : \tau}$ The term t has type τ in context Γ

$$\text{Zero}\ \frac{}{\Gamma \vdash \mathtt{0} : \mathtt{nat}} \qquad \text{Succ}\ \frac{\Gamma \vdash t : \mathtt{nat}}{\Gamma \vdash \mathtt{succ}(t) : \mathtt{nat}} \qquad \text{Pred}\ \frac{\Gamma \vdash t : \mathtt{nat}}{\Gamma \vdash \mathtt{pred}(t) : \mathtt{nat}}$$

$\boxed{\Gamma \vdash t : \tau}$ The term t has type τ in context Γ

$$\textsc{Zero}\ \frac{}{\Gamma \vdash \mathtt{0} : \mathtt{nat}} \qquad \textsc{Succ}\ \frac{\Gamma \vdash t : \mathtt{nat}}{\Gamma \vdash \mathtt{succ}(t) : \mathtt{nat}} \qquad \textsc{Pred}\ \frac{\Gamma \vdash t : \mathtt{nat}}{\Gamma \vdash \mathtt{pred}(t) : \mathtt{nat}}$$

$$\textsc{True}\ \frac{}{\Gamma \vdash \mathtt{true} : \mathtt{bool}} \qquad \textsc{False}\ \frac{}{\Gamma \vdash \mathtt{false} : \mathtt{bool}} \qquad \textsc{IsZ}\ \frac{\Gamma \vdash t : \mathtt{nat}}{\Gamma \vdash \mathtt{zero?}(t) : \mathtt{bool}}$$

$$\textsc{If}\ \frac{\begin{array}{c}\Gamma \vdash b : \mathtt{bool} \\ \Gamma \vdash t : \tau \qquad \Gamma \vdash t' : \tau\end{array}}{\Gamma \vdash \mathtt{if}\ b\ \mathtt{then}\ t\ \mathtt{else}\ t' : \tau}$$

Typing for PCF (II)

$$\textsc{Var}\ \frac{\Gamma(x) = \tau}{\Gamma \vdash x : \tau} \qquad \textsc{Fun}\ \frac{\Gamma, x{:}\sigma \vdash t : \tau}{\Gamma \vdash \mathsf{fun}\, x{:}\sigma.\, t : \sigma \to \tau} \qquad \textsc{App}\ \frac{\Gamma \vdash f : \sigma \to \tau \qquad \Gamma \vdash u : \sigma}{\Gamma \vdash f\ u : \tau}$$

$$\textsc{Fix}\ \frac{\Gamma \vdash f : \tau \to \tau}{\Gamma \vdash \mathtt{fix}(f) : \tau}$$

Typing for PCF (II)

$$\textsc{Var}\ \frac{\Gamma(x) = \tau}{\Gamma \vdash x : \tau} \qquad \textsc{Fun}\ \frac{\Gamma, x{:}\sigma \vdash t : \tau}{\Gamma \vdash \mathtt{fun}\, x{:}\sigma.\, t : \sigma \to \tau} \qquad \textsc{App}\ \frac{\Gamma \vdash f : \sigma \to \tau \qquad \Gamma \vdash u : \sigma}{\Gamma \vdash f\, u : \tau}$$

$$\textsc{Fix}\ \frac{\Gamma \vdash f : \tau \to \tau}{\Gamma \vdash \mathtt{fix}(f) : \tau}$$

$$\mathrm{PCF}_{\Gamma,\tau} \stackrel{\text{def}}{=} \{t \mid \Gamma \vdash t : \tau\} \qquad \mathrm{PCF}_{\tau} \stackrel{\text{def}}{=} \mathrm{PCF}_{\cdot,\tau}$$

PCF

Operational Semantics

Values: $v ::= \underbrace{\texttt{0} \mid \texttt{succ}(v)}_{\underline{n}} \mid \texttt{true} \mid \texttt{false} \mid \texttt{fun}\, x{:}\, \tau.\, t$

Values: $v ::= \underbrace{\mathtt{0} \mid \mathtt{succ}(v)}_{\underline{n}} \mid \mathtt{true} \mid \mathtt{false} \mid \mathtt{fun}\, x{:}\,\tau.\, t$

$$\textsc{Val}\ \frac{\vdash v : \tau}{v \Downarrow_\tau v}$$

PCF evaluation

Values:

$$v ::= \underbrace{\mathtt{0} \mid \mathtt{succ}(v)}_{\underline{n}} \mid \mathtt{true} \mid \mathtt{false} \mid \mathtt{fun}\, x{:}\, \tau.\, t$$

$$\textsc{Val}\ \frac{\vdash v : \tau}{v \Downarrow_\tau v} \qquad \textsc{Succ}\ \frac{t \Downarrow_{\mathtt{nat}} v}{\mathtt{succ}(t) \Downarrow_{\mathtt{nat}} \mathtt{succ}(v)} \qquad \textsc{Pred}\ \frac{t \Downarrow_{\mathtt{nat}} \mathtt{succ}(v)}{\mathtt{pred}(t) \Downarrow_{\mathtt{nat}} v}$$

PCF evaluation

Values: $$v ::= \underbrace{\mathtt{0} \mid \mathtt{succ}(v)}_{\underline{n}} \mid \mathtt{true} \mid \mathtt{false} \mid \mathtt{fun}\, x{:}\,\tau.\, t$$

$$\textsc{Val}\ \frac{\vdash v : \tau}{v \Downarrow_\tau v} \qquad \textsc{Succ}\ \frac{t \Downarrow_{\mathtt{nat}} v}{\mathtt{succ}(t) \Downarrow_{\mathtt{nat}} \mathtt{succ}(v)} \qquad \textsc{Pred}\ \frac{t \Downarrow_{\mathtt{nat}} \mathtt{succ}(v)}{\mathtt{pred}(t) \Downarrow_{\mathtt{nat}} v}$$

$$\textsc{ZeroZ}\ \frac{t \Downarrow_{\mathtt{nat}} \mathtt{0}}{\mathtt{zero?}(t) \Downarrow_{\mathtt{bool}} \mathtt{true}} \qquad \ldots \qquad \textsc{IfT}\ \frac{b \Downarrow_{\mathtt{bool}} \mathtt{true} \qquad t_1 \Downarrow_\tau v}{\mathtt{if}\ b\ \mathtt{then}\ t_1\ \mathtt{else}\ t_2 \Downarrow_\tau v} \qquad \ldots$$

PCF evaluation

Values: $$v ::= \underbrace{\mathtt{0} \mid \mathtt{succ}(v)}_{\underline{n}} \mid \mathtt{true} \mid \mathtt{false} \mid \mathtt{fun}\, x{:}\,\tau.\, t$$

$$\textsc{Val}\ \frac{\vdash v : \tau}{v \Downarrow_\tau v} \qquad \textsc{Succ}\ \frac{t \Downarrow_{\mathtt{nat}} v}{\mathtt{succ}(t) \Downarrow_{\mathtt{nat}} \mathtt{succ}(v)} \qquad \textsc{Pred}\ \frac{t \Downarrow_{\mathtt{nat}} \mathtt{succ}(v)}{\mathtt{pred}(t) \Downarrow_{\mathtt{nat}} v}$$

$$\textsc{ZeroZ}\ \frac{t \Downarrow_{\mathtt{nat}} \mathtt{0}}{\mathtt{zero?}(t) \Downarrow_{\mathtt{bool}} \mathtt{true}} \qquad \dots \qquad \textsc{IfT}\ \frac{b \Downarrow_{\mathtt{bool}} \mathtt{true} \qquad t_1 \Downarrow_\tau v}{\mathtt{if}\ b\ \mathtt{then}\ t_1\ \mathtt{else}\ t_2 \Downarrow_\tau v} \qquad \dots$$

$$\textsc{Fun}\ \frac{t \Downarrow_{\sigma\to\tau} \mathtt{fun}\, x{:}\,\sigma.\, t' \qquad t'[u/x] \Downarrow_\tau v}{t\ u \Downarrow_\tau v} \qquad \textsc{Fix}\ \frac{t\ (\mathtt{fix}(t)) \Downarrow_\tau v}{\mathtt{fix}(t) \Downarrow_\tau v}$$

Values:

$$v ::= \underbrace{\mathtt{0} \mid \mathtt{succ}(v)}_{\underline{n}} \mid \mathtt{true} \mid \mathtt{false} \mid \mathtt{fun}\, x{:}\,\tau.\, t$$

$$\textsc{Val}\ \frac{\vdash v : \tau}{v \Downarrow_\tau v} \qquad \textsc{Succ}\ \frac{t \Downarrow_{\mathtt{nat}} v}{\mathtt{succ}(t) \Downarrow_{\mathtt{nat}} \mathtt{succ}(v)} \qquad \textsc{Pred}\ \frac{t \Downarrow_{\mathtt{nat}} \mathtt{succ}(v)}{\mathtt{pred}(t) \Downarrow_{\mathtt{nat}} v}$$

$$\textsc{ZeroZ}\ \frac{t \Downarrow_{\mathtt{nat}} \mathtt{0}}{\mathtt{zero?}(t) \Downarrow_{\mathtt{bool}} \mathtt{true}} \qquad \dots \qquad \textsc{IfT}\ \frac{b \Downarrow_{\mathtt{bool}} \mathtt{true} \qquad t_1 \Downarrow_\tau v}{\mathtt{if}\ b\ \mathtt{then}\ t_1\ \mathtt{else}\ t_2 \Downarrow_\tau v} \qquad \dots$$

$$\textsc{Fun}\ \frac{t \Downarrow_{\sigma\to\tau} \mathtt{fun}\, x{:}\,\sigma.\, t' \qquad t'[u/x] \Downarrow_\tau v}{t\ u \Downarrow_\tau v} \qquad \textsc{Fix}\ \frac{t\ (\mathtt{fix}(t)) \Downarrow_\tau v}{\mathtt{fix}(t) \Downarrow_\tau v}$$

Alternatively: small-step $t \rightsquigarrow_\tau u$, we have $t \Downarrow_\tau v$ iff $t \rightsquigarrow_\tau^\star u$.

Examples

$$\begin{aligned}\texttt{plus} \overset{\text{def}}{=}\ & \texttt{fun}\, x{:}\,\texttt{nat}.\ \texttt{fix}(\texttt{fun}(p{:}\,\texttt{nat} \to \texttt{nat})(y{:}\,\texttt{nat}).\\ & \quad \texttt{if zero?}(y)\ \texttt{then}\ x\ \texttt{else succ}(p\ \texttt{pred}(y)))\end{aligned}$$

$$\texttt{plus}\ \underline{3}\ \underline{1} \Downarrow_{\texttt{nat}} \underline{4}$$

$$\texttt{plus} \stackrel{\text{def}}{=} \texttt{fun}\, x{:}\, \texttt{nat}.\ \texttt{fix}(\texttt{fun}(p{:}\,\texttt{nat} \to \texttt{nat})(y{:}\,\texttt{nat}).$$
$$\texttt{if zero?}(y)\ \texttt{then}\ x\ \texttt{else}\ \texttt{succ}(p\ \texttt{pred}(y)))$$

$$\texttt{plus}\ \underline{3}\ \underline{1} \Downarrow_{\texttt{nat}} \underline{4}$$

$$\Omega_\tau \stackrel{\text{def}}{=} \texttt{fix}(\texttt{fun}\, x{:}\, \tau.\ x)$$

$$\Omega_\tau \Uparrow_\tau \qquad \text{(diverges)}$$

$$\texttt{plus} \stackrel{\text{def}}{=} \texttt{fun}\, x{:}\,\texttt{nat}.\ \texttt{fix}(\texttt{fun}(p{:}\,\texttt{nat} \to \texttt{nat})(y{:}\,\texttt{nat}).\ \texttt{if zero?}(y)\ \texttt{then}\ x\ \texttt{else succ}(p\ \texttt{pred}(y))))$$

$$\texttt{plus}\ \underline{3}\ \underline{1} \Downarrow_{\texttt{nat}} \underline{4}$$

$$\Omega_\tau \stackrel{\text{def}}{=} \texttt{fix}(\texttt{fun}\, x{:}\,\tau.\ x)$$

$$\Omega_\tau \Uparrow_\tau \qquad \text{(diverges)}$$

Try it out!

PCF is Turing-complete: for every partial recursive function ϕ, there is a PCF term $\underline{\phi}$ such that for all $n \in \mathbb{N}$, if $\phi(n)$ is defined then $\underline{\phi}\ \underline{n} \Downarrow_{\mathsf{nat}} \underline{\phi(n)}$.

Turing-completeness

PCF is Turing-complete: for every partial recursive function ϕ, there is a PCF term $\underline{\phi}$ such that for all $n \in \mathbb{N}$, if $\phi(n)$ is defined then $\underline{\phi}\ \underline{n} \Downarrow_{\mathsf{nat}} \underline{\phi(n)}$.

(Later on: $\phi = \llbracket \underline{\phi} \rrbracket$).

Evaluation in PCF is **deterministic**: if both $t \Downarrow_\tau v$ and $t \Downarrow_\tau v'$ hold, then $v = v'$.

Determinism

Evaluation in PCF is **deterministic**: if both $t \Downarrow_\tau v$ and $t \Downarrow_\tau v'$ hold, then $v = v'$.

By (rule) induction on evaluation $\Downarrow$:

$$\{(t, \tau, v) \mid t \Downarrow_\tau v \wedge \forall v'.(t \Downarrow_\tau v' \Rightarrow v = v')\}$$

Intuition: there is always exactly one rule which applies.

PCF

Contextual equivalence

Contextual equivalence – informal

Two phrases of a programming language are **contextually equivalent** if any occurrences of the first phrase in a **complete program** can be replaced by the second phrase without affecting the **observable results** of executing the program.

Two phrases of a programming language are **contextually equivalent** if any occurrences of the first phrase in a **complete program** can be replaced by the second phrase without affecting the **observable results** of executing the program.

The intuitive notion of **program equivalence** for programmers.

Evaluation contexts

$$
\begin{aligned}
\mathcal{C} \quad ::= \quad & - \mid \mathtt{succ}(\mathcal{C}) \mid \mathtt{pred}(\mathcal{C}) \mid \mathtt{zero?}(\mathcal{C}) \mid \\
& \mathtt{if}\ \mathcal{C}\ \mathtt{then}\ t\ \mathtt{else}\ t \mid \mathtt{if}\ t\ \mathtt{then}\ \mathcal{C}\ \mathtt{else}\ t \mid \mathtt{if}\ t\ \mathtt{then}\ t\ \mathtt{else}\ \mathcal{C} \mid \\
& \mathtt{fun}\, x{:}\tau.\, \mathcal{C} \mid \mathcal{C}\ t \mid t\ \mathcal{C} \mid \mathtt{fix}(\mathcal{C})
\end{aligned}
$$

$$\begin{array}{rcl}\mathcal{C} & ::= & - \mid \mathtt{succ}(\mathcal{C}) \mid \mathtt{pred}(\mathcal{C}) \mid \mathtt{zero?}(\mathcal{C}) \mid \\ & & \mathtt{if}\ \mathcal{C}\ \mathtt{then}\ t\ \mathtt{else}\ t \mid \mathtt{if}\ t\ \mathtt{then}\ \mathcal{C}\ \mathtt{else}\ t \mid \mathtt{if}\ t\ \mathtt{then}\ t\ \mathtt{else}\ \mathcal{C} \mid \\ & & \mathtt{fun}\, x{:}\tau.\,\mathcal{C} \mid \mathcal{C}\ t \mid t\ \mathcal{C} \mid \mathtt{fix}(\mathcal{C})\end{array}$$

Typing extended to evaluation contexts: $\Gamma \vdash_{\Delta,\sigma} \mathcal{C} : \tau$.

$$\mathcal{C} \quad ::= \quad - \mid \texttt{succ}(\mathcal{C}) \mid \texttt{pred}(\mathcal{C}) \mid \texttt{zero?}(\mathcal{C}) \mid$$
$$\texttt{if}\ \mathcal{C}\ \texttt{then}\ t\ \texttt{else}\ t \mid \texttt{if}\ t\ \texttt{then}\ \mathcal{C}\ \texttt{else}\ t \mid \texttt{if}\ t\ \texttt{then}\ t\ \texttt{else}\ \mathcal{C} \mid$$
$$\texttt{fun}\ x{:}\tau.\ \mathcal{C} \mid \mathcal{C}\ t \mid t\ \mathcal{C} \mid \texttt{fix}(\mathcal{C})$$

Typing extended to evaluation contexts: $\Gamma \vdash_{\Delta,\sigma} \mathcal{C} : \tau$.

$$\frac{}{\Gamma \vdash_{\Gamma,\tau} - : \tau} \qquad \frac{\Gamma \vdash_{\Delta,\sigma} \mathcal{C} : \tau_1 \to \tau_2 \qquad \Gamma \vdash u : \tau_1}{\Gamma \vdash_{\Delta,\sigma} \mathcal{C}\ u : \tau_2} \qquad \dots$$

Contextual equivalence

Given a type τ, a typing context Γ and terms $t, t' \in \mathrm{PCF}_{\Gamma,\tau}$, contextual equivalence, written $\Gamma \vdash t \cong_{\mathrm{ctx}} t' : \tau$ is defined to hold if for all evaluation contexts $\mathcal{C}$ such that $\cdot \vdash_{\Gamma,\tau} \mathcal{C} : \gamma$, where γ is `nat` or `bool`, and for all values $v \in \mathrm{PCF}_{\gamma}$,

$$\mathcal{C}[t] \Downarrow_{\gamma} v \Leftrightarrow \mathcal{C}[t'] \Downarrow_{\gamma} v.$$

When Γ is the empty context, we simply write $t \cong_{\mathrm{ctx}} t' : \tau$ for $\cdot \vdash t \cong_{\mathrm{ctx}} t' : \tau$.

PCF

Introducing denotational semantics

The aims of denotational semantics

- a mapping of PCF types τ to domains $[\![\tau]\!]$;
- a mapping of closed, well-typed PCF terms $\cdot \vdash t : \tau$ to elements $[\![t]\!] \in [\![\tau]\!]$;
- denotation of open terms will be continuous functions.

THE AIMS OF DENOTATIONAL SEMANTICS

- a mapping of PCF types τ to domains $[\![\tau]\!]$;
- a mapping of closed, well-typed PCF terms $\cdot \vdash t : \tau$ to elements $[\![t]\!] \in [\![\tau]\!]$;
- denotation of open terms will be continuous functions.

Compositionality: $[\![t]\!] = [\![t']\!] \Rightarrow [\![\mathcal{C}[t]]\!] = [\![\mathcal{C}[t']]\!]$.

Soundness: for any type τ, $t \Downarrow_\tau v \Rightarrow [\![t]\!] = [\![v]\!]$.

Adequacy: for $\gamma = \texttt{bool}$ or $\texttt{nat}$, if $t \in \mathrm{PCF}_\gamma$ and $[\![t]\!] = [\![v]\!]$ then $t \Downarrow_\gamma v$.

Proof principle: to show

$$t_1 \cong_{\text{ctx}} t_2 : \tau$$

it suffices to establish

$$[\![t_1]\!] = [\![t_2]\!] \in [\![\tau]\!]$$

Proof principle: to show

$$t_1 \cong_{\mathrm{ctx}} t_2 : \tau$$

it suffices to establish

$$[\![t_1]\!] = [\![t_2]\!] \in [\![\tau]\!]$$

$$\begin{aligned}
\mathcal{C}[t_1] \Downarrow_{\mathtt{nat}} v &\Rightarrow [\![\mathcal{C}[t_1]]\!] = [\![v]\!] && \text{(soundness)} \\
&\Rightarrow [\![\mathcal{C}[t_2]]\!] = [\![v]\!] && \text{(compositionality on } [\![t_1]\!] = [\![t_2]\!]\text{)} \\
&\Rightarrow \mathcal{C}[t_2] \Downarrow_{\mathtt{nat}} v && \text{(adequacy)}
\end{aligned}$$

Proof principle: to show

$$t_1 \cong_{\text{ctx}} t_2 : \tau$$

it suffices to establish

$$[\![t_1]\!] = [\![t_2]\!] \in [\![\tau]\!]$$

$$\begin{aligned} \mathcal{C}[t_1] \Downarrow_{\mathtt{nat}} v &\Rightarrow [\![\mathcal{C}[t_1]]\!] = [\![v]\!] && \text{(soundness)} \\ &\Rightarrow [\![\mathcal{C}[t_2]]\!] = [\![v]\!] && \text{(compositionality on } [\![t_1]\!] = [\![t_2]\!]\text{)} \\ &\Rightarrow \mathcal{C}[t_2] \Downarrow_{\mathtt{nat}} v && \text{(adequacy)} \end{aligned}$$

and symmetrically for $\mathcal{C}[t_2] \Downarrow_{\mathtt{nat}} v \Rightarrow \mathcal{C}[t_1] \Downarrow_{\mathtt{nat}} v$, and similarly for `bool`.

The power of denotational semantics

Proof principle: to show

$$t_1 \cong_{\text{ctx}} t_2 : \tau$$

it suffices to establish

$$[\![t_1]\!] = [\![t_2]\!] \in [\![\tau]\!]$$

Denotational equality is **sound**, but is it **complete**?
Does equality in the model imply contextual equivalence?

Proof principle: to show

$$t_1 \cong_{\mathrm{ctx}} t_2 : \tau$$

it suffices to establish

$$[\![t_1]\!] = [\![t_2]\!] \in [\![\tau]\!]$$

Denotational equality is sound, but is it complete?
Does equality in the model imply contextual equivalence?

Full abstraction.

Denotational Semantics for PCF

Denotational Semantics for PCF

Types and contexts

$$[\![\texttt{nat}]\!] \stackrel{\text{def}}{=} \mathbb{N}_\perp \qquad \text{(flat domain)}$$

$$[\![\texttt{bool}]\!] \stackrel{\text{def}}{=} \mathbb{B}_\perp \qquad \text{(flat domain)}$$

$$[\![\tau \to \tau']\!] \stackrel{\text{def}}{=} [\![\tau]\!] \to [\![\tau']\!] \qquad \text{(function domain)}$$

$$[\![\Gamma]\!] \stackrel{\text{def}}{=} \prod_{x \in \mathrm{dom}(\Gamma)} [\![\Gamma(x)]\!] \qquad (\Gamma\text{-environments})$$

Semantics of contexts

$$[\![\Gamma]\!] \stackrel{\text{def}}{=} \prod_{x \in \text{dom}(\Gamma)} [\![\Gamma(x)]\!] \qquad (\Gamma\text{-environments})$$

- $[\![\cdot]\!] = \mathbb{1}$ (one element set)
- $[\![x{:}\,\tau]\!] = (\{x\} \to [\![\tau]\!]) \cong [\![\tau]\!]$
- $[\![x_1{:}\,\tau_1, \ldots, x_n{:}\,\tau_n]\!] = [\![\tau_1]\!] \times \cdots \times [\![\tau_n]\!]$

Denotational Semantics for PCF

Terms

To every typing judgement

$$\Gamma \vdash t : \tau$$

we associate a continuous function

$$[\![\Gamma \vdash t : \tau]\!] : [\![\Gamma]\!] \to [\![\tau]\!]$$

between domains. In other words,

$$[\![-]\!] : \mathrm{PCF}_{\Gamma,\tau} \to [\![\Gamma]\!] \to [\![\tau]\!]$$

Denotation of operations on $\mathbb{B}$ and $\mathbb{N}$

$$\begin{aligned} \text{succ} : \quad \mathbb{N} &\to \mathbb{N} \\ n &\mapsto n + 1 \end{aligned}$$

$$\begin{aligned} \text{pred} : \quad \mathbb{N} &\to \mathbb{N} \\ 0 &\mapsto \mathsf{undefined} \\ n + 1 &\mapsto n \end{aligned}$$

$$\begin{aligned} \text{zero?} : \quad \mathbb{N} &\to \mathbb{B} \\ 0 &\mapsto \text{true} \\ n + 1 &\mapsto \text{false} \end{aligned}$$

Denotation of operations on $\mathbb{B}$ and $\mathbb{N}$

$$
\begin{aligned}
\mathrm{succ}_\perp : \quad \mathbb{N}_\perp &\to \mathbb{N}_\perp \\
n &\mapsto n+1 \\
\perp &\mapsto \perp
\end{aligned}
$$

$$
\begin{aligned}
\mathrm{pred}_\perp : \quad \mathbb{N}_\perp &\to \mathbb{N}_\perp \\
0 &\mapsto \perp \\
n+1 &\mapsto n \\
\perp &\mapsto \perp
\end{aligned}
$$

$$
\begin{aligned}
\mathrm{zero?}_\perp : \quad \mathbb{N}_\perp &\to \mathbb{B}_\perp \\
0 &\mapsto \mathrm{true} \\
n+1 &\mapsto \mathrm{false} \\
\perp &\mapsto \perp
\end{aligned}
$$

Denotation of operations on $\mathbb{B}$ and $\mathbb{N}$

$$\begin{aligned}
[\![\texttt{0}]\!](\rho) &\stackrel{\text{def}}{=} 0 && \in \mathbb{N}_\perp \\
[\![\texttt{true}]\!](\rho) &\stackrel{\text{def}}{=} \text{true} && \in \mathbb{B}_\perp \\
[\![\texttt{false}]\!](\rho) &\stackrel{\text{def}}{=} \text{false} && \in \mathbb{B}_\perp
\end{aligned}$$

Denotation of operations on $\mathbb{B}$ and $\mathbb{N}$

$$
\begin{aligned}
[\![\texttt{0}]\!]\,(\rho) &\overset{\text{def}}{=} 0 && \in \mathbb{N}_\bot \\
[\![\texttt{true}]\!]\,(\rho) &\overset{\text{def}}{=} \text{true} && \in \mathbb{B}_\bot \\
[\![\texttt{false}]\!]\,(\rho) &\overset{\text{def}}{=} \text{false} && \in \mathbb{B}_\bot \\
\\
[\![\texttt{succ}(t)]\!]\,(\rho) &\overset{\text{def}}{=} \text{succ}_\bot([\![t]\!]\,(\rho)) && \in \mathbb{N}_\bot \\
[\![\texttt{pred}(t)]\!]\,(\rho) &\overset{\text{def}}{=} \text{pred}_\bot([\![t]\!]\,(\rho)) && \in \mathbb{N}_\bot \\
[\![\texttt{zero?}(t)]\!]\,(\rho) &\overset{\text{def}}{=} \text{zero?}_\bot([\![t]\!]\,(\rho)) && \in \mathbb{B}_\bot
\end{aligned}
$$

$$[\![\texttt{succ}(t)]\!] = \text{succ}_\bot \circ [\![t]\!]$$

Denotation of operations on $\mathbb{B}$ and $\mathbb{N}$

$$
\begin{aligned}
[\![\texttt{0}]\!](\rho) &\overset{\text{def}}{=} 0 && \in \mathbb{N}_\perp \\
[\![\texttt{true}]\!](\rho) &\overset{\text{def}}{=} \text{true} && \in \mathbb{B}_\perp \\
[\![\texttt{false}]\!](\rho) &\overset{\text{def}}{=} \text{false} && \in \mathbb{B}_\perp \\
\\
[\![\texttt{succ}(t)]\!](\rho) &\overset{\text{def}}{=} \text{succ}_\perp([\![t]\!](\rho)) && \in \mathbb{N}_\perp \\
[\![\texttt{pred}(t)]\!](\rho) &\overset{\text{def}}{=} \text{pred}_\perp([\![t]\!](\rho)) && \in \mathbb{N}_\perp \\
[\![\texttt{zero?}(t)]\!](\rho) &\overset{\text{def}}{=} \text{zero?}_\perp([\![t]\!](\rho)) && \in \mathbb{B}_\perp \\
\\
[\![\texttt{if}\ b\ \texttt{then}\ t\ \texttt{else}\ t']\!] &\overset{\text{def}}{=} \text{if}([\![b]\!](\rho), [\![t]\!](\rho), [\![t']\!](\rho)) && \in [\![\tau]\!]
\end{aligned}
$$

$$
[\![\texttt{if}\ b\ \texttt{then}\ t\ \texttt{else}\ t']\!] = \text{if} \circ \langle [\![b]\!], \langle [\![t]\!], [\![t']\!] \rangle\rangle
$$