

# Typesetting examples in the style of Pierce's book

Andrew Tolmach

January 8, 2008

I use an ad-hoc collection of tools for typesetting:

- For representing object language syntax, I define macros in the style shown above, extending the set as necessary to cover new language features. This is tedious, but it helps me keep things well-formed.
- For stating inference rules, I use the `bcprules` package.
- For showing derivations, I use the `proof` package.

For example, here are the inference rules in Figures 8-1 and 8-2 on p.93:

$$\text{true} : \text{Bool} \qquad (\text{T-TRUE})$$
$$\text{false} : \text{Bool} \qquad (\text{T-FALSE})$$
$$\frac{t_1 : \text{Bool} \quad t_2 : T \quad t_3 : T}{\text{if } t_1 \text{ then } t_2 \text{ else } t_3 : T} \qquad (\text{T-IF})$$
$$0 : \text{Nat} \qquad (\text{T-ZERO})$$
$$\frac{t_1 : \text{Nat}}{\text{succ } t_1 : \text{Nat}} \qquad (\text{T-SUCC})$$
$$\frac{t_1 : \text{Nat}}{\text{pred } t_1 : \text{Nat}} \qquad (\text{T-PRED})$$
$$\frac{t_1 : \text{Nat}}{\text{iszero } t_1 : \text{Bool}} \qquad (\text{T-ISZERO})$$

Here is the text of Lemma 8.2.2 on p. 94:

1. If `true` : R, then R = Bool.
2. If `false` : R, then R = Bool.
3. If `if t1 then t2 else t3` : R, then t<sub>1</sub> : Bool, t<sub>2</sub> : R, and t<sub>3</sub> : R.
4. If `0` : R, then R = Nat.
5. If `succ t1` : R, then R = Nat and t<sub>1</sub> : Nat.
6. If `pred t1` : R, then R = Nat and t<sub>1</sub> : Nat.
7. If `iszero t1` : R, then R = Bool and t<sub>1</sub> : Nat.

Finally, here is the derivation near the bottom of p. 94:

$$\frac{\frac{\overline{0 : \text{Nat}} \text{ T-ZERO}}{\text{iszero } 0 : \text{Bool}} \text{ T-ISZERO} \quad \frac{\overline{0 : \text{Nat}} \text{ T-ZERO}}{\text{pred } 0 : \text{Nat}} \text{ T-PRED}}{\text{if iszero } 0 \text{ then } 0 \text{ else pred } 0 : \text{Nat}} \text{ T-IF}$$

I'm sure there are better ways of doing some or all of these things; contributions and suggestions are welcome!