

115 This question explores a simpler, more elegant function theory than the one presented in Chapter 3. We separate the notion of local variable introduction from the notion of domain, and we generalize the latter to become local axiom introduction. Variable introduction has the form $\langle v \cdot b \rangle$ where v is a variable and b is any expression (the body; no domain). There is an Application Law

$$\langle v \cdot b \rangle x = (\text{substitute } x \text{ for } v \text{ in } b)$$

and an Extension Law

$$f = \langle v \cdot f v \rangle$$

Let a be binary, and let b be any expression. Then $a \gg b$ is an expression of the same type as b . The \gg operator has precedence level 12 and is right-associating. Its axioms include:

$$\top \gg b = b$$

$$a \gg b \gg c = a \wedge b \gg c$$

The expression $a \gg b$ is a “one-tailed if-expression”, or “asserted expression”; it introduces a as a local axiom within b . A function is a variable introduction whose body is an asserted expression in which the assertion has the form $v: D$. In this case, we allow an abbreviation: for example, the function $\langle n: n: \text{nat} \gg n+1 \rangle$ can be abbreviated $\langle n: \text{nat} \cdot n+1 \rangle$. Applying this function to 3, we find

$$\begin{aligned} & \langle n: n: \text{nat} \gg n+1 \rangle 3 \\ &= 3: \text{nat} \gg 3+1 \\ &= \top \gg 4 \\ &= 4 \end{aligned}$$

Applying it to -3 we find

$$\begin{aligned} & \langle n: n: \text{nat} \gg n+1 \rangle (-3) \\ &= -3: \text{nat} \gg -3+1 \\ &= \perp \gg -2 \end{aligned}$$

and then we are stuck; no further axiom applies. In the example, we have used variable introduction and axiom introduction together to give us back the kind of function we had; but in general, they are independently useful.

- (a) Show how function-valued variables can be introduced in this new theory.
- (b) What expressions in the old theory have no equivalent in the new? How closely can they be approximated?
- (c) What expressions in the new theory have no equivalent in the old? How closely can they be approximated?

no solution given