(3.3 #6) Is 3 a sequence number? What is \( \text{lh} 3 \)? Find \((1 \ast 3) \ast 6\) and \(1 \ast (3 \ast 6)\).

**Solution:** By the definitions on pp. 220-221, \( b \) is a sequence number if and only if for some \( m \geq -1 \) and some \( a_0, \ldots, a_m \),

\[
b = \langle a_0, \ldots, a_m \rangle = p_0^{a_0+1} \cdots p_m^{a_m+1},
\]

where \( p_i \) is the \((i+1)\)st prime.

To write 3 as a sequence number, it must be expressed as a product of prime powers with positive powers.

\[
3 = 2^0 \cdot 3^1.
\]

Since the power of 2 is zero (and by the unique factorization of integers), \( 3 \neq \langle a_0, \ldots, a_m \rangle \) for any \( a_0, \ldots, a_m \).

Nonetheless, the function \( \text{lh} \) is still defined on 3:

\[
\text{lh} 3 = \text{least } n \text{ such that either } 3 = 0 \text{ or } p_n \text{ does not divide } 3 = \text{least } n \text{ such that } p_n \text{ does not divide } 3.
\]

To find this \( n \), try each one in turn: \( n = 0 \) works if \( p_0 \mid 3 \). Since \( p_0 = 2 \) and \( 2 \mid 3 \), we have \( \text{lh} 3 = 0 \).

The concatenation function is also defined for all numbers:

\[
(1 \ast 3) \ast 6 = \left( 1 \prod_{i < \text{lh} 3} p_{i+\text{lh} 3}^{(3)_{i+1}} \right) \ast 6 = 1 \ast 6 = 1 \prod_{i < \text{lh} 6} p_{i+\text{lh} 6}^{(6)_{i+1}} = \prod_{i < 2} p_{i}^{(6)_{i+1}} = p_0^{1} p_1^{1} = 2 \cdot 3 = 6.
\]

In the above, we used that \( \text{lh} 3 = 0 \) and so the first product is empty and hence by convention is equal to 1. Also, we used that \( 6 = \langle 0, 0 \rangle \) so \( \text{lh} 6 = 2 \) and \( (6)_0 = 0 = (6)_1 \).

Finally, we note that \( \text{lh} 1 = 0 \) (by a similar calculation to \( \text{lh} 3 = 0 \)).

\[
1 \ast (3 \ast 6) = 1 \ast \left( 3 \prod_{i < \text{lh} 6} p_{i+\text{lh} 6}^{(6)_{i+1}} \right) = 1 \ast \left( 3 \prod_{i < 2} p_{i}^{(6)_{i+1}} \right) = 1 \ast (3 \cdot p_0^{1} p_1^{1}) = 1 \ast 18
\]

\[
= 1 \cdot \prod_{i < \text{lh} 18} p_{i+\text{lh} 18}^{(18)_{i+1}} = 1 \cdot p_0^{1} p_1^{2} = 18.
\]

In this case, we use that \( 18 = 2^1 \cdot 3^2 = \langle 0, 1 \rangle \) so \( \text{lh} 18 = 2 \), \( (18)_0 = 0 \), \( (18)_1 = 1 \).

Notice that these calculations imply that \( \ast \) is not associative in general.

(However, the book comments that if we restrict the domain of \( \ast \) to the sequence numbers, we get an associative function.)

(3.3 #9) Show that there is a representable function \( f \) such that for every \( n, a_0, \ldots, a_n \)

\[f(\langle a_0, \ldots, a_n \rangle) = a_n.\]

**Solution:** Recall the definition on p. 212: a function \( f : \mathbb{N} \to \mathbb{N} \) is representable (in \( Cn \ A_E \)) if and only if there is a formula \( \varphi \) with free variables \( v_1, v_2 \) such that for every \( a_1 \in \mathbb{N} \)

\[A_E \vdash \forall v_2 (\varphi(S^{a_1}0, v_2) \iff v_2 = S^{f(a_1)0}).\]

In this case, we are formalizing the function

\[f(a) = b \quad \text{iff} \quad a \text{ is a sequence number, and } \text{lh } a = n + 1 \text{ for some } n \text{, and } (a)_n = b.\]
Let $\varphi_{\text{seq}}(v_1)$ be the wff representing the set of sequence numbers, let $\varphi_{\text{lh}}(v_1, v_2)$ be the wff representing the length function, and let $\varphi_{\text{proj}}(v_1, v_2, v_3)$ be the wff representing the decoding function. Consider the wff

$$\varphi_{\text{seq}}(v_1) \land \exists x \left( \varphi_{\text{lh}}(v_1, Sx) \land \varphi_{\text{proj}}(v_1, x, v_2) \right).$$

In fact, the quantifier $\exists x$ can be bounded because the length of a sequence number is no more than the number. Therefore, define $\varphi(v_1, v_2)$ to be the wff

$$\varphi_{\text{seq}}(v_1) \land \exists x \left( x < v_1 \land \varphi_{\text{lh}}(v_1, Sx) \land \varphi_{\text{proj}}(v_1, x, v_2) \right).$$

Then $\varphi$ represents $f$ because it defines the graph of $f$ and is numeralwise determined by $A_E$.

(1.7 #11)

(a) Explain why the union of two effectively enumerable sets is effectively enumerable.

*Solution:* If either of the sets is finite then the result is clear: simply enumerate all the elements in the finite set first and then proceed to enumerate the second set. Otherwise, assume that both sets are infinite. We can interleave the procedures that effectively enumerate the two sets. That is, suppose procedure $P_1$ enumerates $A_1$ and $P_2$ enumerates $A_2$. Then run $P_1$ until it outputs some number, and output it. Then pause $P_1$ and run $P_2$ until it outputs some number, and output it. Keep switching between the two procedures in this manner, outputting each number that is produced by either one of the procedures. The collection of outputs will be exactly $A_1 \cup A_2$.

(b) Explain why the intersection of two effectively enumerable sets is effectively enumerable.

*Solution:* This procedure is similar to part (b) except that we hold back on outputting a number until we’ve seen it appear in both lists. We can think of keeping the set of outputs so far from each of $A_1, A_2$ on a “scratchwork” piece of memory and when an output from one of them appears that is already on this list of previously seen elements, output it.