COMPSCI/SFWRENG 2FA3
Discrete Mathematics with Applications II
Winter 2020

Week 03 Exercises

Dr. William M. Farmer
McMaster University

Revised: January 17, 2020

Exercises

1. Let $\mathsf{FinSeq}_{\mathbb{N}}$ be the set of finite sequences whose members are in $\mathbb{N}$.

   a. Define $\mathsf{FinSeq}_{\mathbb{N}}$ as an inductive set.

   b. Use an auxiliary function, recursion, and pattern matching to define the function

   $$\mathsf{reverse} : \mathsf{FinSeq}_{\mathbb{N}} \to \mathsf{FinSeq}_{\mathbb{N}}$$

   such that $\mathsf{reverse}(s)$ is the reverse of s for all $s \in \mathsf{FinSeq}_{\mathbb{N}}$.

   c. Write the structural induction principle for $\mathsf{FinSeq}_{\mathbb{N}}$.

2. Let Nat be the natural numbers defined as an inductive set in the lecture notes, $\mathbb{B}$ be the set of boolean values true and false, $\mathsf{odd} : \mathsf{Nat} \to \mathbb{B}$ be the function that maps the odd natural numbers to true and the even natural numbers to false, and $\mathsf{even} : \mathsf{Nat} \to \mathbb{B}$ be the function that maps the even natural numbers to true and the odd natural numbers to false. Define odd and even simultaneously by pattern matching using "mutual recursion".

3. Let $\mathsf{BinTree}$ be the inductive set and nodes and ht be the functions defined in the lecture notes. Let $\mathsf{leaves} : \mathsf{BinTree} \to \mathbb{N}$ be the function that maps a binary to the number of leaf nodes in it.

   a. Define leaves by pattern matching and recursion.

   b. Prove that, for all $t \in \mathsf{BinTree}$,

   $$\mathsf{leaves}(t) \leq 2^{\mathsf{ht}(t)}$$

   by structural induction.

4. Let $\mathsf{BinTree}$ be the inductive set defined in the lecture notes. Let $\mathsf{mirror} : \mathsf{BinTree} \to \mathsf{BinTree}$ be the function that maps a binary tree to its "mirror image".

   a. Define mirror by pattern matching and recursion.

   b. Prove that, for all $t \in \mathsf{BinTree}$,

   $$\mathsf{mirror}(\mathsf{mirror}(t)) = t$$

   by structural induction.

5. Let $\mathsf{BinTree}$ be the inductive set defined in the lectures. A *subtree* of $t \in \mathsf{BinTree}$ is t itself or a subcomponent of t that is a member of $\mathsf{BinTree}$.

   a. Define a function $\mathsf{subtrees} : \mathsf{BinTree} \to \mathsf{set}(\mathsf{BinTree})$ that maps each $t \in \mathsf{BinTree}$ to the set of subtrees of t.

   b. Prove by structural induction that, if $t \in \mathsf{BinTree}$ contains n Branch nodes, then t has at most $2n + 1$ subtrees.

6. Let S be the set of bit strings defined inductively by:

   a. $\texttt{"0"} \in S$.

   b. If $s \in S$, then $\texttt{"0"} + s \in S$ and $s + \texttt{"0"} \in S$.

   c. If $s \in S$, then , $\texttt{"0"} + s + \texttt{"1"} \in S$ and $\texttt{"1"} + s + \texttt{"0"} \in S$.

   $s + t$ denotes the concatenation of s and t. Prove by structural induction that, for all strings $s \in S$, the number of 1s in s is less than or equal to the number of 0s in s.

7. Suppose $(S_1, \leq_1)$ and $(S_2, \leq_2)$ are weak partial orders. Prove that $(S_1 \times S_2, \leq)$ is a weak partial order where $(s_1, s_2) \leq (s'_1, s'_2)$ iff $s_1 \leq_1 s'_1$ and $s_2 \leq_2 s'_2$.

8. Let $<_{\text{lex}} \subseteq (\mathbb{N} \times \mathbb{N}) \times (\mathbb{N} \times \mathbb{N})$ be lexicographical order, i.e.,

   $$(x_1, y_1) <_{\text{lex}} (x_2, y_2)$$

   iff $x_1 < x_2$ or ($x_1 = x_2$ and $y_1 < y_2$).

   a. Prove that $(\mathbb{N} \times \mathbb{N}, <_{\text{lex}})$ is a well-order.

   b. Write the ordinal induction principle for $(\mathbb{N} \times \mathbb{N}, <_{\text{lex}})$.

   c. Prove by the ordinal induction principle for $(\mathbb{N} \times \mathbb{N}, <_{\text{lex}})$ that A, the version of the Ackermann function presented in the lecture notes, is defined on all members of $\mathbb{N} \times \mathbb{N}$.

9. Let $(S, <)$ be a partial order such that S is finite. Prove that $(S, <)$ is well-founded.

10. Let $(\mathbb{N}, R_{\mathsf{suc}})$ be the mathematical structure where

    $$m \; R_{\mathsf{suc}} \; n \text{ iff } n = m + 1.$$

    Prove that $(\mathbb{N}, R_{\mathsf{suc}})$ is well-founded.

11. The Ackermann function was originally defined as the ternary function $B : \mathbb{N} \times \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ such that:

    a. $B(m, n, 0) = m + n$.
    b. $B(m, 0, 1) = 0$.
    c. $B(m, 0, 2) = 1$.
    d. $B(m, 0, p) = m$ for $p > 2$.
    e. $B(m, n, p) = B(m, B(m, n - 1, p), p - 1)$ for $n > 0$ and $p > 0$.

    Prove that B is defined on all members of $\mathbb{N} \times \mathbb{N} \times \mathbb{N}$ using well-founded induction.