Assignment: Section 5.3: 12, 26c ("5 | a+b" meas that 5 divides a+b), 43, 44; ( 7th edition)

In Exercises 12–19 *fn* is the *n*th Fibonacci number.

**12.** Prove that *f* 2 1+ *f* 22 +· · ·+*f* 2 *n* = *fnfn*+1 when *n* is a positive integer.

**P(n) = *f* 2 1+ *f* 22 +· · ·+*f* 2 *n* = *fnfn*+1**

**Basis Step: f1 x f2 = 1 x 1 = 1**

**Recursive Step: Assuming P(n) is true, show P(n+1) is also true.**

**This means *f* 2 1+ *f* 22 +· · ·+*f* 2 *n* +*f* 2 *n+1* = *fn*+1*fn*+2**

**= *fnfn*+1 + f2n+2**

**= *fn*+1(*fn +fn+1*)**

**=*fn+1fn+2***

**26.** Let *S* be the subset of the set of ordered pairs of integers defined recursively by

*Basis step: (*0*,* 0*)* ∈ *S*.

*Recursive step:* If *(a, b)* ∈ *S*, then *(a* + 2*, b* + 3*)* ∈ *S* and *(a* + 3*, b* + 2*)* ∈ *S*.

**c)** Use structural induction to show that 5 | *a* + *b* when *(a, b)* ∈ *S*.

**By the basis step we know that 5| 0 + 0 = 0.**

**Recursive step: Show if 5|a + b, then 5|a+b for any elements obtained from a,b.**

**If a+b = 5k for an int k, where (a+2, b +3) then a+b +5 = 5k + 5 = 5(k+1), where k+1 is also an int.**

**In the same way, for (a + 3, b +2) = a + b + 5 = 5k + 5 = 5(k+1) where k + 1 is also an int.**

**43.** Use structural induction to show that *n(T )* ≥ 2*h(T )* + 1, where *T* is a full binary tree, *n(T )* equals the number of vertices of *T* , and *h(T )* is the height of *T* . The set of leaves and the set of internal vertices of a full binary tree can be defined recursively.

*Basis step:* The root *r* is a leaf of the full binary tree with exactly one vertex *r*. This tree has no internal vertices.

*Recursive step:* The set of leaves of the tree *T* = *T*1 · *T*2 is the union of the sets of leaves of *T*1 and of *T*2. The internal vertices of *T* are the root *r* of *T* and the union of the set of internal vertices of *T*1 and the set of internal vertices of *T*2.

**Basis Step is true because n(T) = 1 and h(T) = 0, and 1 >= 2 x 0 + 1 (when a tree has just a root node)**

**Recursive Step: Show n(T) >= 2h(T)+1 for the full binary tree T.**

**T is formed by two subtrees, T1 and T2 + the root node (where T1 and T2 are smaller than T) by the recursive def of a full binary tree.**

**T1 and T2 is holds true, and by the recursive definition of n(T) and h(T), we know that n(T) = 1 + n (T1 ) + n(T2) and h(T) = 1 + max(h(T1), h(T2))**

**n(T) = 1 + n(T1) + n(T2)**

**>= 1 + 2h(T1) + 1 + 2h(T2) + 1 Inductive Hyp**

**>= 1+2max(h(T1), h(T2)) + 2 By 2h(T1) + 2h(T2) >= 2max(h(T1), h(T2))**

**= 1 + 2(max(hT1), h(T2)) + 1) Factor**

**= 1 + 2h(T) Recursive Def of FBT**

**44.** Use structural induction to show that *l(T )*, the number of leaves of a full binary tree *T* , is 1 more than *i(T )*, the number of internal vertices of *T* .

**Basis Step: The smallest full binary tree is single root r: l(T) = 1 = 1 + i(T)**

**Recursive Step: Show result holds for T**

**T is formed by two subtrees, T1 and T2 + the root node (where T1 and T2 are smaller than T) by the recursive def of a full binary tree**

**We know by the basis step that T1 and T2 hold and that i(T) = i(T1 ) + i(T2 ) + 1**

**l(T) = l(T1 ) + l (T2 ) Rec Def of Full Binary Tree**

**= i (T1 ) + 1 + i( T2 ) + 1 Induction Hypoth**

**= i(T) + 1 Recursive Def of FBT**

**Completing the inductive step**