Strong Induction Problems.

Boolean BinSearch ( int key, SortedListofIntegers s )
if the length of s is zero
    return false
else if the length of s is one
    return true if the list is key otherwise return false
else let k be the middle element of s, s1 the list before k and s2 the list after.
    if key is k
        return true
    else if key < k
        return BinSearch ( key, s1 )
    else
        return BinSearch ( key, s2 )

1. Prove by strong induction on the length of the list s that BinSearch halts.
2. Assuming BinSearch halts, prove by strong induction on the length of the list s that BinSearch correctly determines if key is in the list.

List MergeSort ( List s )
if the length of the list is less than or equal 1
    return s
else divide the list into halves s1 and and s2 (or as near halves as possible)
    return Merge( MergeSort(s1), MergeSort(s2) )

3. Assuming Merge halts, prove by strong induction on the length of the list s that MergeSort halts.
4. Assuming MergeSort halts and Merge is correct, prove by strong induction on the length of the list s that MergeSort returns a sorted list.

VeryLongInteger Multiply ( VeryLongInter x, VeryLongInteger y)
Let n be the number of digits in the longest of x and y, and d = n/2.
If n less than or equal 1
    return x * y
else find x1,x2,y1,y2 each have no more than d digits so that
    x = x1 \cdot 10^d + x2 and y = y1 \cdot 10^d + y2
let m1 = Multiply ( x1, y1 )
let m2 = Multiply ( x2, y2 )
let m3 = Multiply ( x1 + x2, y1 + y2 ) - m1 - m2
return m1 \cdot 10^{2d} + m3 \cdot 10^d + m2

5. Prove by strong induction on the number of digits n that Multiply halts.
6. Assuming Multiply halts, prove by strong induction on the number of digits n that Multiply correctly multiplies the two numbers x and y.