Homework 11: Due Friday, April 10

Problem 1: Prove the following by structural induction on binary trees: If t is a binary tree with n levels, then t has at most $2^n - 1$ many nodes.

Problem 2: From Problem 1 and discussion class, it appears that a reasonably balanced binary tree with n levels has approximately 2^n many nodes. For example, if t is perfectly balanced, then it would have 1 node on the first level, 2 nodes on the second level, $4 = 2^2$ many nodes on the third level, $8 = 2^3$ many nodes on fourth level, etc. Thus, to approximate the number of nodes of a binary tree with n leavels, we would want to evaluate the sum

$$1 + 2 + 2^2 + 2^3 + \cdots + 2^{n-1}.$$

Instead of working with this directly, we evaluate a more general sum. Let $r \in \mathbb{R}$ with $r \neq 1$. Prove that

$$1 + r + r^2 + r^3 + \cdots + r^n = \frac{r^{n+1} - 1}{r - 1}$$

by induction on $n \in \mathbb{N}$.

Problem 3: Write a Scheme function bin-tree-map that takes two inputs, a function f from integers to integers and a binary tree t, and returns the binary tree resulting from applying f to every node of t.

Problem 4: Suppose that we code sets of integers using binary search trees. Write a Scheme function bst-add-elem that takes two inputs, an integer a and a binary search tree t, and returns a binary search tree obtained by adding a to t.

Problem 5: In each part of this problem, write a paragraph explaining why your code works.
a. Write a recursive Scheme function split that takes two inputs, a natural number k and a list as, and returns a list of length three '(bs d cs) where bs is the list consisting of the first k elements of as, d is the next element of as, and cs is the list consisting of the remaining elements of as. For example,

```
(split 2 '(0 3 4 8 10 17))
```

should return

```
'((0 3) 4 (8 10 17)).
```

You may assume that $0 \leq$ k $<$ (length as).
b. Write a recursive Scheme function that takes as input an ordered list of integers (in increasing order), and returns a balanced binary search tree with the same elements. If you need to make it slightly unbalanced, put an extra node on the left subtree instead of the right. For example, on input '(1 2 3 4), your function should output

```
'(3 (2 (1 () ()) ()) (4 () ())).
```