## **Concepts and Definitions**

- 1. What is mutual recursion?
- 2. When working with trees, what is (typically) the input to each mutually recursive procedure?
- 3. What is a binary tree? Which selectors do we use for it?
- 4. What is car/cdr recursion?

## Working with Trees

1. Write a procedure add-child-lengths which takes as input a tree, and produces as output a new tree in which each datum is now a list whose first element is the original datum, and whose second element is the number of children it has. (You can find the length of a list using the length procedure.) Use mutual recursion here.

2. Write sum-tree, which takes as input a tree of numbers and produces as output hte sum of all the numbers in the tree. You should use sequence operations (map, filter, accumulate) to implement it. No helper procedures allowed! (This is a hard question to start with, so you may want to first do it with mutual recursion, which is easier.)

3. Write listify-tree, which takes as input a tree and produces as output a list of all of the datums in the tree, in any order.

4. Write count-leaves, which returns the number of leaves in the input tree. A leaf is any tree which has no children.

## Car/cdr Recursion

1. Write the procedure sum-binary-tree, which sums the datum of a binary tree.

2. Now, write the procedure sum-cons-structure. Would (sum-cons-structure bt) evaluate to the same thing as (sum-binary-tree bt) (assuming that bt is a valid input to sum-binary-tree)? If yes, why don't we generally do this? If no, why not?