Recursion Problems: Group A

The following problems all involve recursion, memoization, or dynamic programming. Try to see if you can come up with the most efficient solutions possible!

1. Generate all strings of n pairs of balanced parentheses. For example, if n = 3, you'd generate the strings ((())), (()()), (())(), ()(()), ()()().

2. You are given a pyramid of numbers like the one shown here:

   137

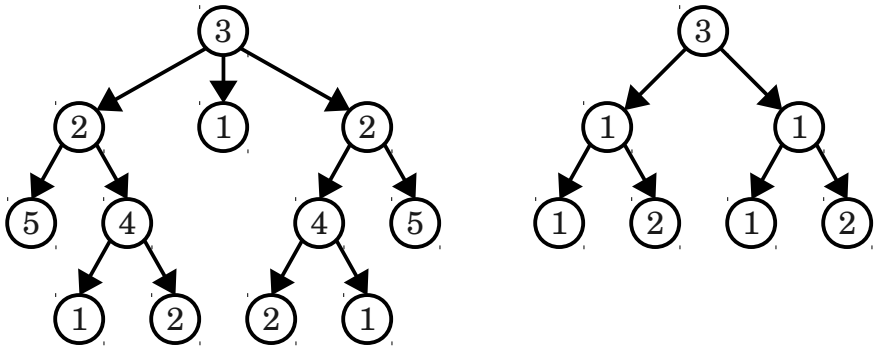
   42 -15

   -4 13 45

   21 14 -92 33

   Values in the pyramid can be both positive or negative. A path from the top of the pyramid to the bottom consists of starting at the top of the pyramid and taking steps diagonally left or diagonally right down to the bottom of the pyramid. The cost of a path is the sum of all the values in the pyramid. Find the path from the top of the pyramid to the bottom with the highest total cost.

3. A *palindromic tree* is a tree that is the same when it's mirrored around the root. For example, the left tree below is a palindromic tree and the right tree below is not:



   Given a tree, determine whether it is a palindromic tree.

4. The Fibonacci strings are a series of recursively-defined strings. F_0 is the string **a**, F_1 is the string **bc**, and F_{n+2} is the concatenation of F_n and F_{n+1}. For example, F_2 is **abc**, F_3 is **bcabc**, F_4 is **abcbcabc**, etc. Given a number n and an index k, return the kth character of the string F_n.