Problem Statement
The goal of this assignment is to get acquainted with the programming language Prolog!

Requirements Analysis
This assignment has two parts (worth 67% and 33% of the grade, respectively). For the first part Patrick needs a count the number of symmetric trees of a given height, while for the second Sandy wants to explore the versatility of the append/3 relation.

Design: Symmetric Trees
A tree can be represented in Prolog as a collection of tree nodes where each node is given as t(value, left-subtree, right-subtree). For example, t(v, t(v, nil, t(v, nil, nil)), t(v, t(v, nil, nil), nil)).

Apply the generate-and-test paradigm to construct all symmetric and balanced binary trees with a given number of nodes. For example
?- trees(5,Ts).
Ts = [t(x, t(x, nil, t(x, nil, nil)), t(x, t(x, nil, nil), nil)),
     t(x, t(x, t(x, nil, nil), nil), t(x, nil, t(x, nil, nil)))]

How many such trees are there with 57 nodes? Investigate about how many solutions there are for a given number of nodes? What if the number is even?

Design: Universal append/3
1) Using only the append/3 relation, formulate queries to determine the following.
   a) The third element of a list.
   b) The last element of a list.
   c) All but the last element of a list.
   d) Whether a list Y is formed by inserting an element A somewhere in a list X.
   e) If list L2 is composed of every other element from list L1.
      For this one you may use every_other(A,B) on the right-hand side of the rule.
2) Then produce three tests for each.

What to hand in
Upload your code and output as the four files square.pl, square.out (collected using script(1)), appends.pl, and appends.out Please edit script’s output to cleanup any unwanted control characters, and to annotate and explain your program’s output.

Notes
(1) As a warmup consider writing the prolog goal count_leaves/2 to count the leaves of a binary tree. A leaf is a node with no successors.
(2) Man script(1).
(3) There exist solutions to these problems on the web. I fully believe that you have mastered the zen of downloading “stuff” off the web. Your goal here is not to practice this zen art, but to learn some Prolog. To that end please avoid Mr. Internet once you start writing your code.