# Lab 10 — Tree-based Dynamic Programming

CS/ECE 374 B

October 30, 2019

For these questions, you are given a tree as a linked data structure. Each tree node has a list of children. You may add one or more fields to the node data structure to help your algorithm.

1. Describe a recursive algorithm to find the maximum independent set of a tree. Your algorithm should take a root node as its input and output the size of the maximum independent set of the tree. (Unlike some past problems, you should not need to add any extra arguments to the recursive function, i.e., your recursive calls should all be of the form MIS(n) for some tree node n).

2. Design and analyze a dynamic programming version of your algorithm. Hint: store the intended result in node.mis

3. Consider now that each node has a weight, node.w. Design an algorithm for finding (the weight of) the maximum *weight* independent set.

4. Your goal in this problem is to assign each node one of three labels, 1, 2, or 3. Each node *must* have a different label than its parent. The *cost* of an assignment is the number red nodes that have a label smaller than their parent. For example, in the figure below, the red nodes are ones that have a label than their parent, and therefore the cost of this labeling would be 9.
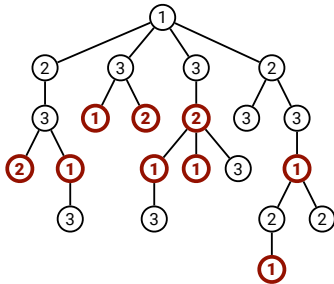
Your goal is to find the (cost of) the minimal cost labeling of a given tree. (The labeling above is *not* the minimal cost one.)