Maximize independent node weights on binary tree

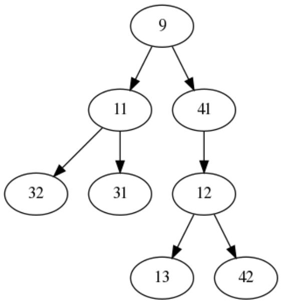
Serge Kruk

November 4, 2021

1 Problem statement

Imagine a binary rooted tree with a positive weight on each node. We want to choose a subset of the nodes so that the total sum of the weights is maximized with the constraint that no two adjacent nodes can be chosen.

For instance, in the following tree, the set of nodes with weights $(41, 13, 42, 32, 31)$ is optimal. Note that no two nodes are adjacent.



- Hint 0: Solve this recursively. Then add-on memoization.
- Hint 1: How can we solve this recursively? One important observation is that in any rooted binary tree, the optimal list of nodes that doesn't include the root node will be the union of optimal lists for the left and right subtrees. And the optimal list that does include the root node will be the union of optimal lists for the left and right subtrees that do not include their respective root nodes.