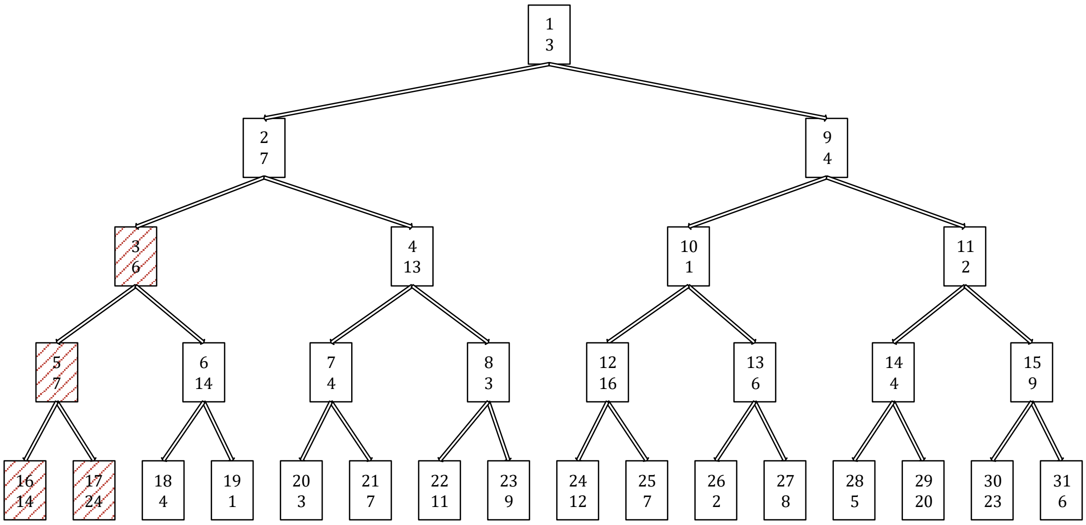
Traverse Binary Tree DP

遍历二叉树动规

问题：

在一个二叉树上，从任意节点i到达另一个任意节点j的路线是唯一的。假设该二叉树上的每个节点都是一个牧场，而每个牧场中都有一只奶牛，节点i的奶牛拥有一个权值，为v_i，奶牛会在二叉树上游荡，但它游荡的位置不超过一个距离，为$Dist$。节点i的牧场上拥有的奶牛数量并不固定，可能拥有 0 只奶牛，那么节点i拥有的权值为 0；可能拥有n只奶牛，那么节点i拥有的权值为这n只奶牛的权值之和，即$\sum_1^n v_k$（其中$k \in [1, n]$）。求出所有节点中权值最大的节点的权值。

对于下图中的二叉树，每个节点的标号为上面的数字，权值为下面的数字。但奶牛游荡的距离为 1 时，会拥有最大权值的节点为节点 5，最大权值为$51 = 6 + 7 + 14 + 24$，即节点 3、5、16、17 的权值之和：

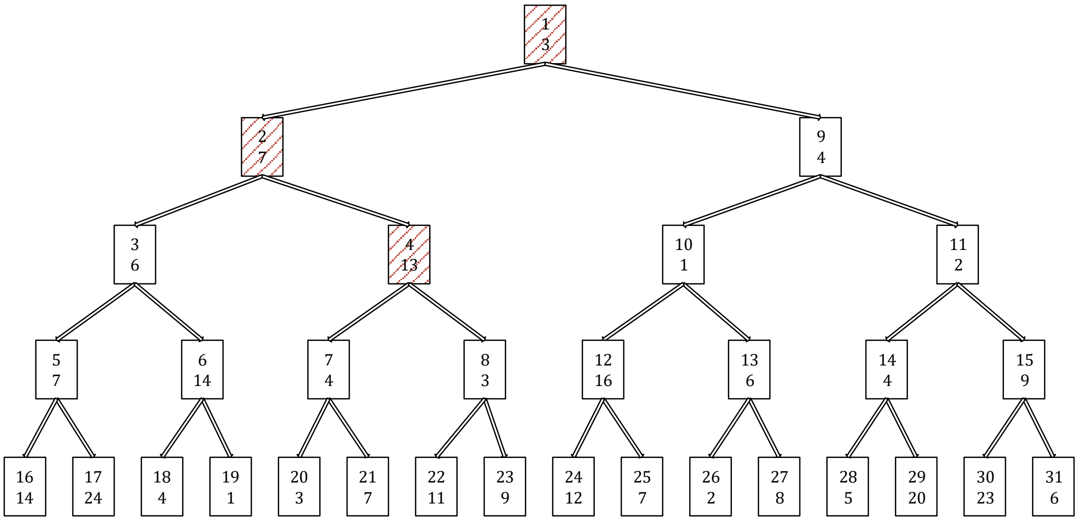


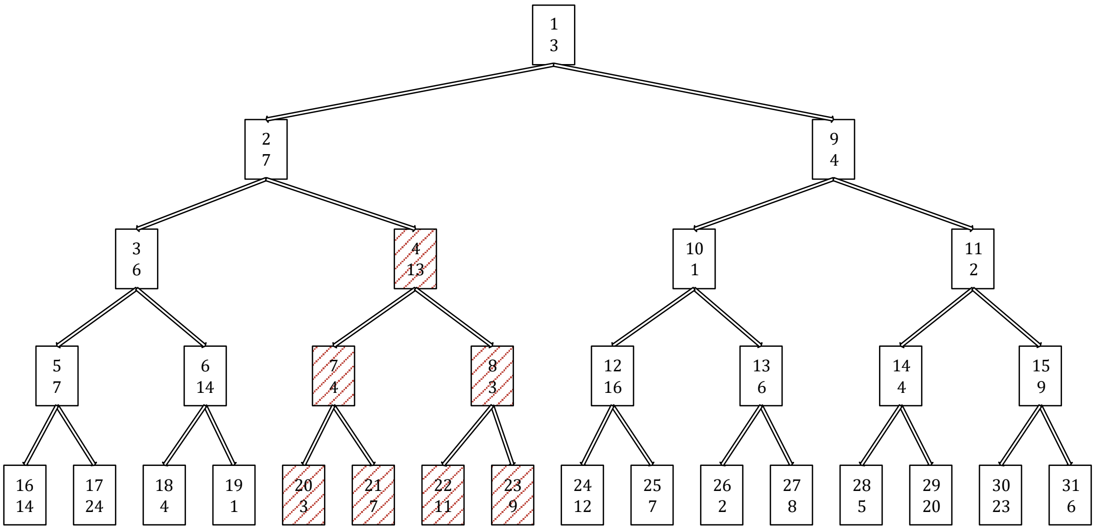
本问题的原型为 USACO Mar 2008 “Cow Travelling”（游荡的奶牛）。

解法：

问题中的示例是一种最简单的情况，即游荡距离为 1，这时节点i的最大权值即为该节点与所有相邻节点的权值之和。当游荡距离增大，节点i的最大权值为所有到该节点的距离不超过$Dist$的节点的权值之和，即$\sum_1^n v_k$（其中n为所有节点的数量，k节点的权值为v_k）。

节点i可以到达二叉树的向上和向下$Dist$层的所有节点，如图：





对于节点 4 的奶牛，当其$Dist = 2$，则向上可以到达节点 2、1；向下可以到达 7、8、20、21、22、23。

(1)向上可达的所有节点的权值和：

设$up(i,j)$为游荡距离为j的节点i，向上可以到达的所有节点的权值之和，则有$up(i,j) = up(father_i, j-1) + v_{father-i}$，即游荡距离为$j$的节点$i$，其向上可达的权值和，等于游荡距离为$j-1$的父节点$father_i$的向上可达的权值和与父节点自己的权值之和。在上图中可以看出，游荡距离为 2 的节点 4，其向上权值和$up(4,2)$，恰好等于游荡距离为 1 的节点 2 的向上权值和与节点 2 的权值之和，即$up(4,2) = up(2,1) + v_2$。

对于游荡距离为 0 的节点i，其向上权值和为$up(i,0) = 0$。

(2)向下可达的所有节点的权值和：

设$down(i,j)$为游荡距离为 j 的节点 i，向下可以到达的所有节点的权值之和，则有$down(i,j) = down(leftChild_i, j-1) + down(rightChild_i, j-1) + v_{leftChild-i} + v_{rightChild-i}$，即游荡距离为$j$的节点$i$，其向下可达的权值和，等于游荡距离为$j-1$的左右孩子节点$leftChild_i$和$rightChild_i$的向下可达权值和，与左右孩子节点自己的权值的总和。在上图中可以看出，游荡距离为 2 的节点 4，其向下权值和$down(4,2)$，恰好等于游荡距离为 1 的节点 7、8 的向下权值和，与节点 7、8 的权值的总和，即$down(4,2) = down(7,1) + down(8,1) + v_7 + v_8$。

对于游荡距离为 0 的节点i，其向下权值和为$down(i,0) = 0$。

根据(1)和(2)两个部分，可以得出游荡距离为j的节点i的最大权值$f(i,j) = up(i,j) + down(i,j) + v_i$。

最终在所有$f(i,j)$中选择最大值作为返回结果（其中$i,j \in [0,n)$）。该算法的时间复杂度是$O(n \times n)$。

USACO Mar 2008 “Cow Travelling”（游荡的奶牛）：

http://train.usaco.org/TESTDATA/MAR08.ctravel.htm