Exercise 1. Tree Conversions. Consider the Harris walk of an ordered tree on \(n\) nodes, i.e., a binary sequence of length \(2(n - 1)\) denoting the direction of movement when one walks around a tree. From this walk, given in array form, give a linear time algorithm that reconstructs the tree in oldest child / next sibling pointer form.

Exercise 2. Recursive Programs. In what follows, \(T\) is a binary tree. Nodes are denoted by \(u\) and \(v\), and the root is \(r\). Left and right children of \(u\) are denoted by \(\text{left}[u]\) and \(\text{right}[u]\). Each node has an integer “value” \(\text{val}[u]\). Write top-down \(O(n)\) time recursive programs that compute, for all nodes \(u\), the attributes \(\text{height}[u]\), \(\text{max}[u]\) and \(\text{sum}[u]\), where,

(i) \(\text{height}[u]\) is the maximal distance from \(u\) to any leaf in its subtree.
(ii) \(\text{max}[u]\) is a pointer to the ancestor of \(u\) that has the maximal value (among ancestors of \(u\)).
(iii) \(\text{sum}[u]\) is the sum all the values of the nodes in the subtree of \(u\).

Exercise 3. Preorder and Postorder. An expression tree is a rooted ordered tree in which each internal node corresponds to an operator, which can be binary (+, -, *, etcetera) or unary (negate, cos, exp, sin, tan, etcetera), and each leaf is an operand (a number, say). Its postorder listing is called a postfix expression. An example is

\[
ab + cd - \ast e \cos +
\]

which represents the expression

\[
((a + b) \ast (c + d)) + \cos(e).
\]

Given an array with a postfix listing of length \(n\), find an \(O(n)\) time algorithm that outputs a prefix listing for the same expression. You may assume that for each item in the listing, it is known whether it is a binary operator, a unary operator, or an operand. No recursions please: if necessary, use stacks in an explicit manner.