Example: Expression Trees

In-order traversal of this tree gives back the expression (where * and / take precedence over + and –); pre-order traversal changes the expression to prefix notation (the operator comes before the operands) and post-order traversal gives postfix notation (the operator comes after the operands).
Evaluate an expression tree with in-order traversal:

function EVAL(v)
  if v=nil then return 0
  else if v is a leaf then return the value stored at v
  else begin
    OP := the operator stored at v
    return OP(EVAL(LCHILD(v)),EVAL(RCHILD(v)))
  end
end

(expression trees continued)
Note: An arithmetic expression can be converted to an expression tree in linear time. The tree provides a nice data structure to evaluate the expression for different values of the variables.