An application of binary trees:

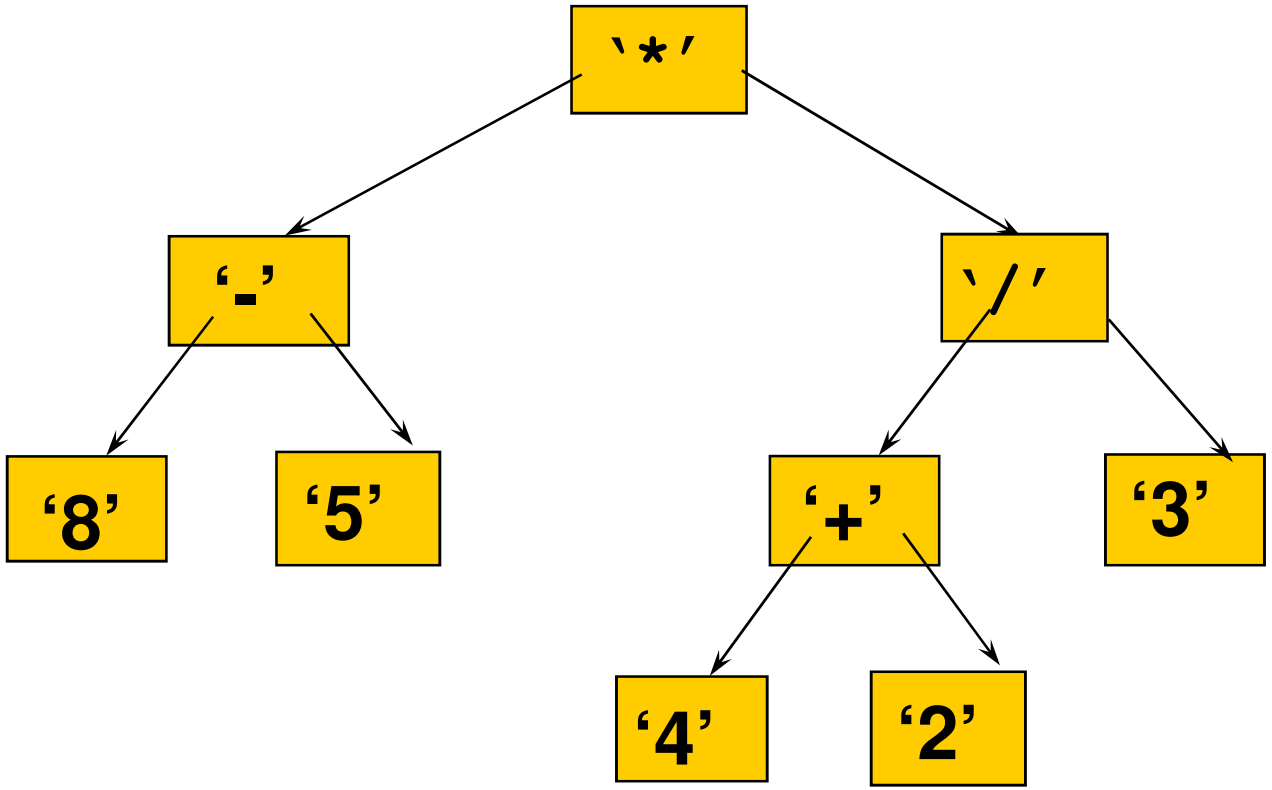
Binary Expression Trees

A Binary Expression Tree is . . .

A special kind of binary tree in which:

1. **Each leaf node contains a single operand**
2. **Each nonleaf node contains a single binary operator**
3. **The left and right subtrees of an operator node represent subexpressions that must be evaluated before applying the operator at the root of the subtree.**

A Four-Level Binary Expression



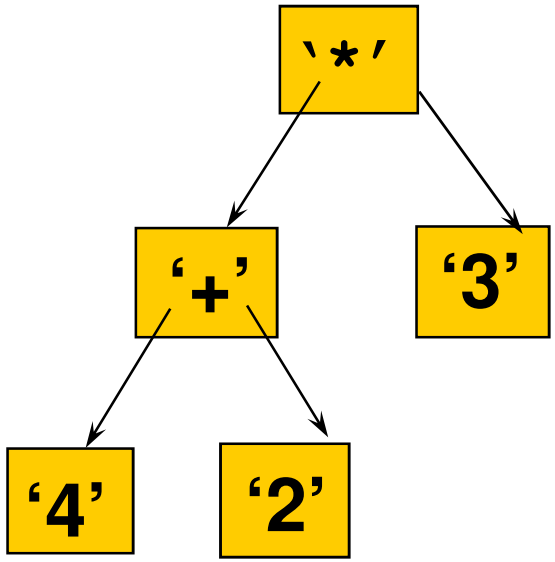
Levels Indicate Precedence

The levels of the nodes in the tree indicate their relative precedence of evaluation (we do not need parentheses to indicate precedence).

Operations at higher levels of the tree are evaluated later than those below them.

The operation at the root is always the last operation performed.

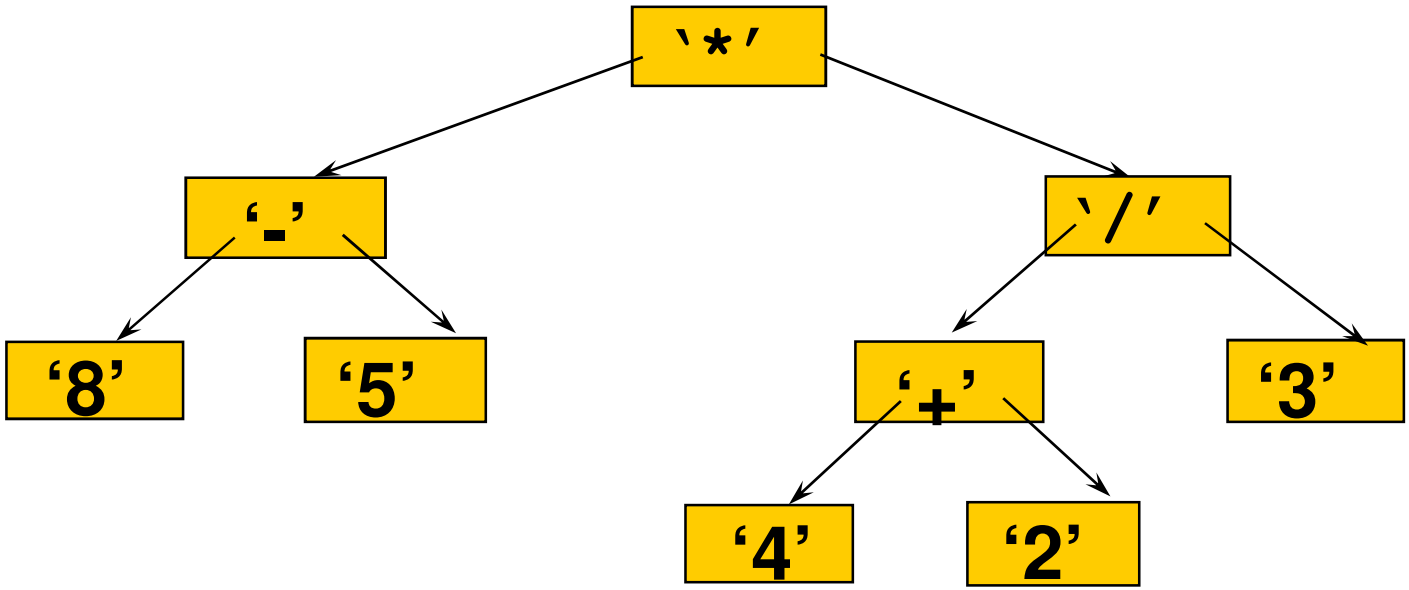
A Binary Expression Tree



What value does it have?

(4 + 2) * 3 = 18

Easy to generate the infix, prefix, postfix expressions (how?)

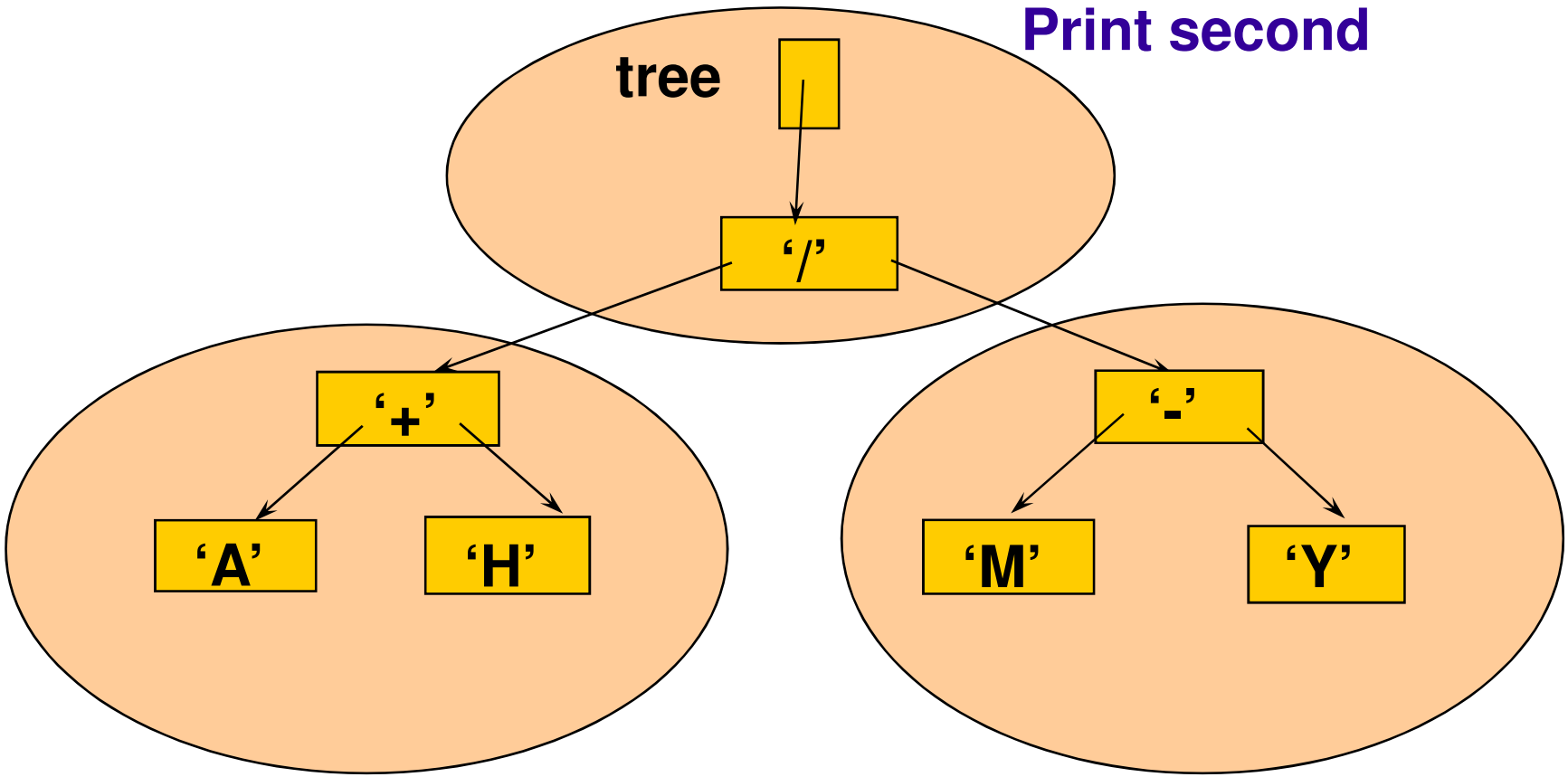


Infix: ((8 - 5) * ((4 + 2) / 3))

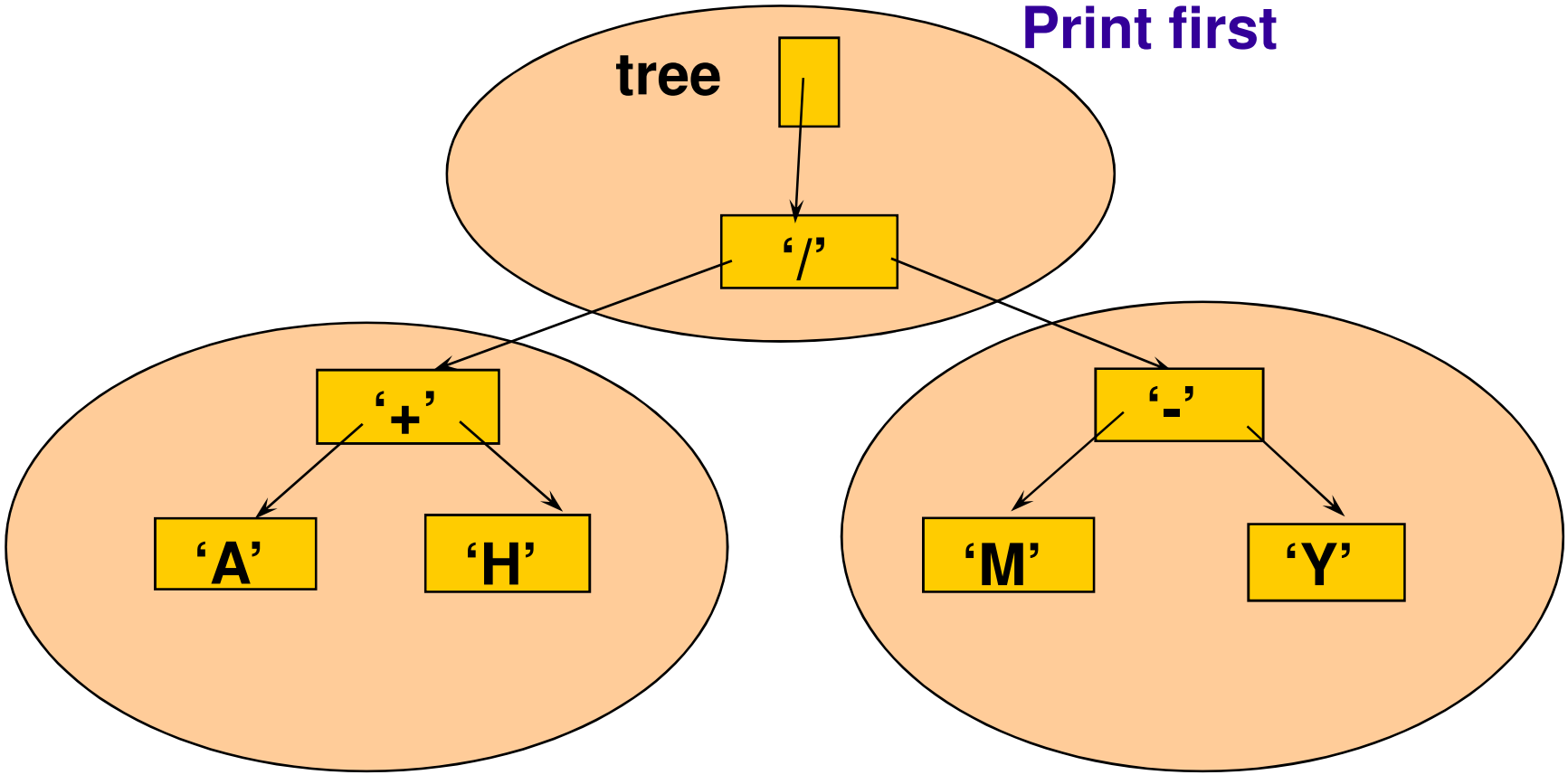
Prefix: * - 8 5 / + 4 2 3

Postfix: 8 5 - 4 2 + 3 / *

Inorder Traversal: (A + H) / (M - Y)



Preorder Traversal: / + A H - M Y



Postorder Traversal: A H + M Y - /