CSE-101: Discrete Mathematics

Chapter 11: Trees

Lec Md Jakaria

Dept of CSE, MIST

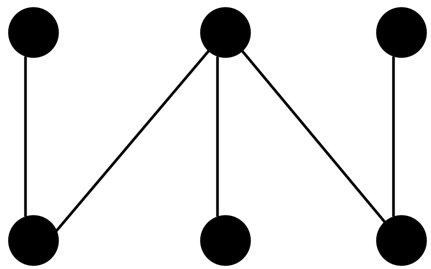
Tree

Definition 1. **A tree is a connected undirected graph with no simple circuits.**

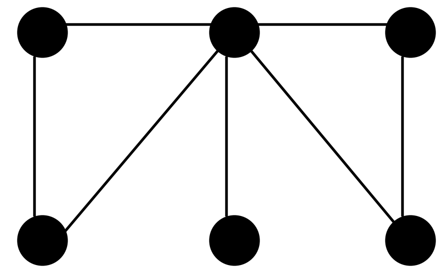
Theorem 1. **An undirected graph is a tree if and only if there is a unique simple path between any two of its vertices.**

Which graphs are trees?

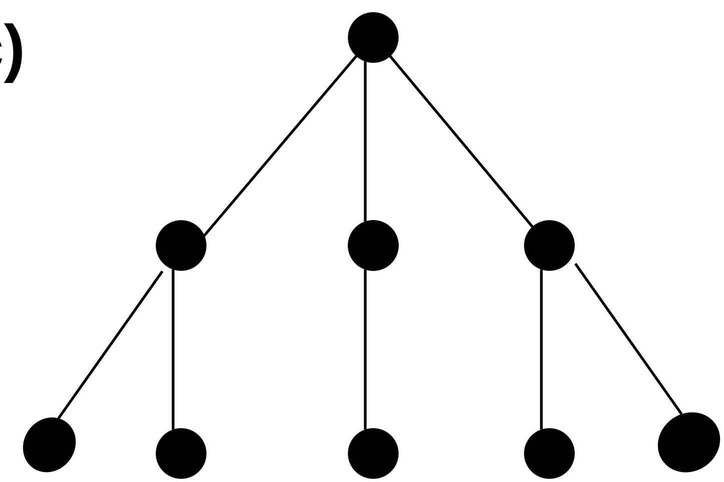
a)



b)



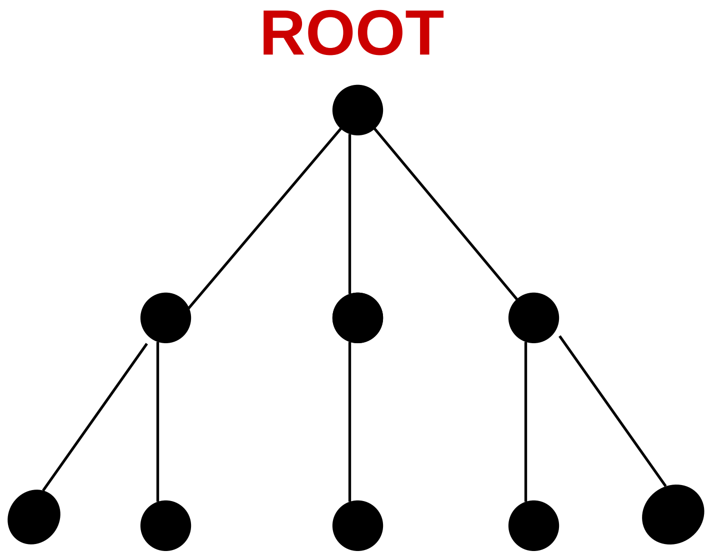
c)



Specify a vertex as root

Then, direct each edge away from the root.

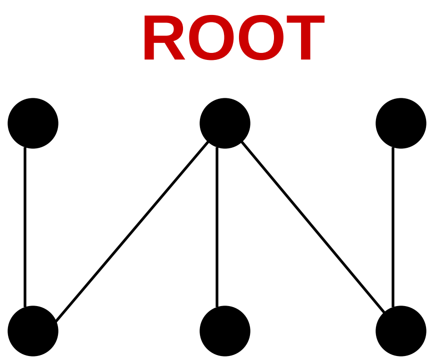
c)



Specify a root.

Then, direct each edge away from the root.

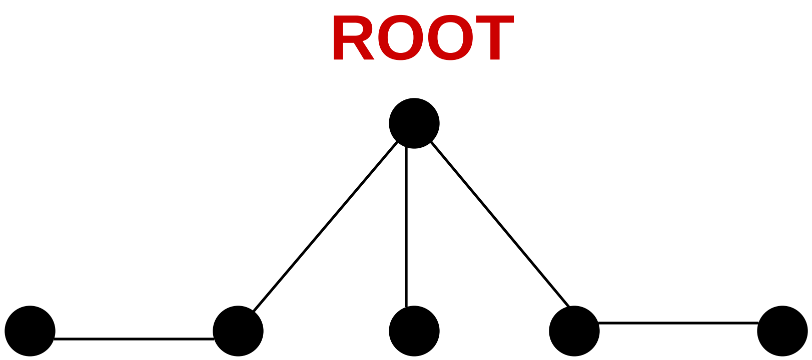
a)



Specify a root.

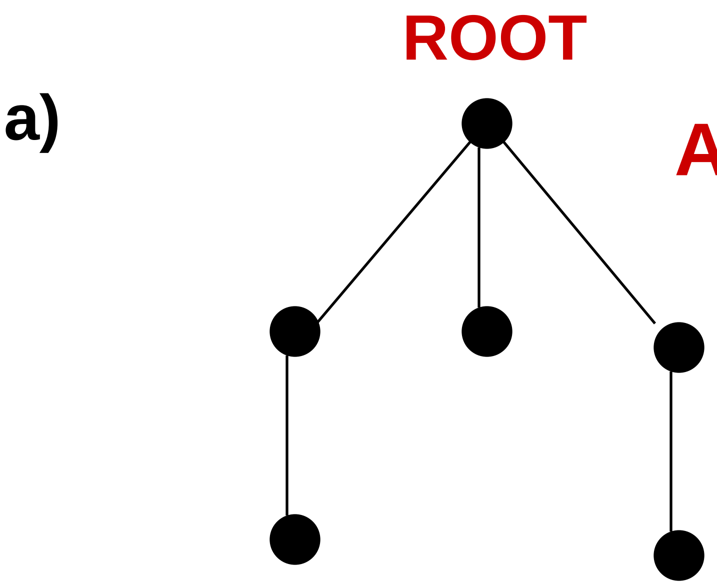
Then, direct each edge away from the root.

a)



Specify a root.

Then, direct each edge away from the root.

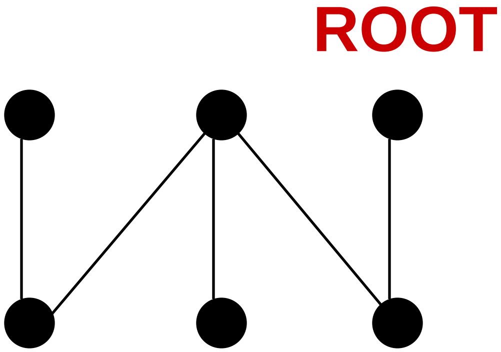


A directed graph called a rooted tree results.

What if a different root is chosen?

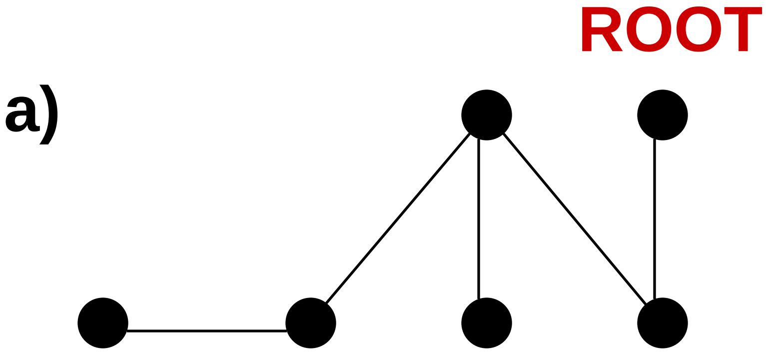
Then, direct each edge away from the root.

a)



What if a different root is chosen?

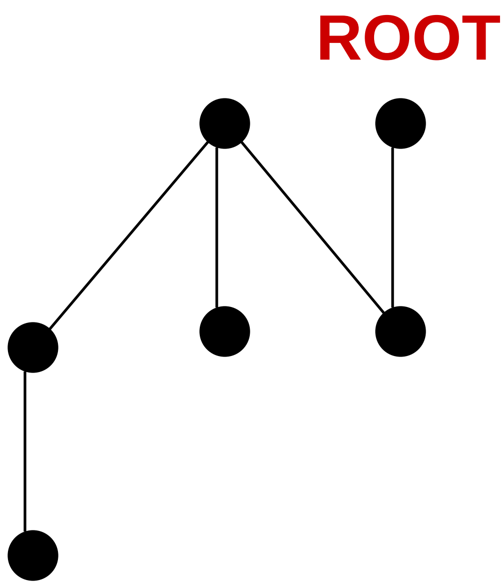
Then, direct each edge away from the root.



What if a different root is chosen?

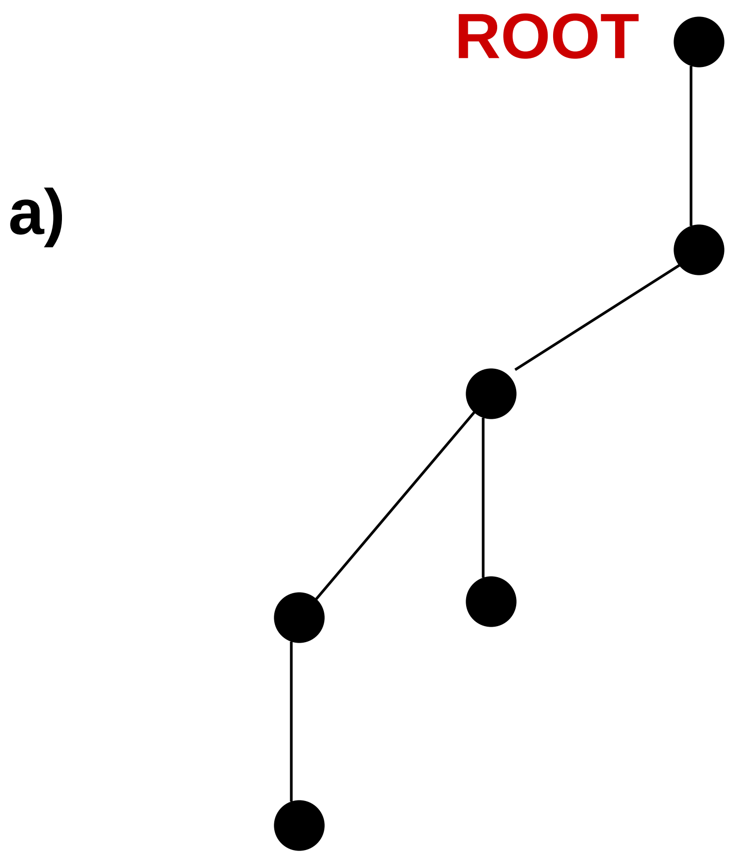
Then, direct each edge away from the root.

a)



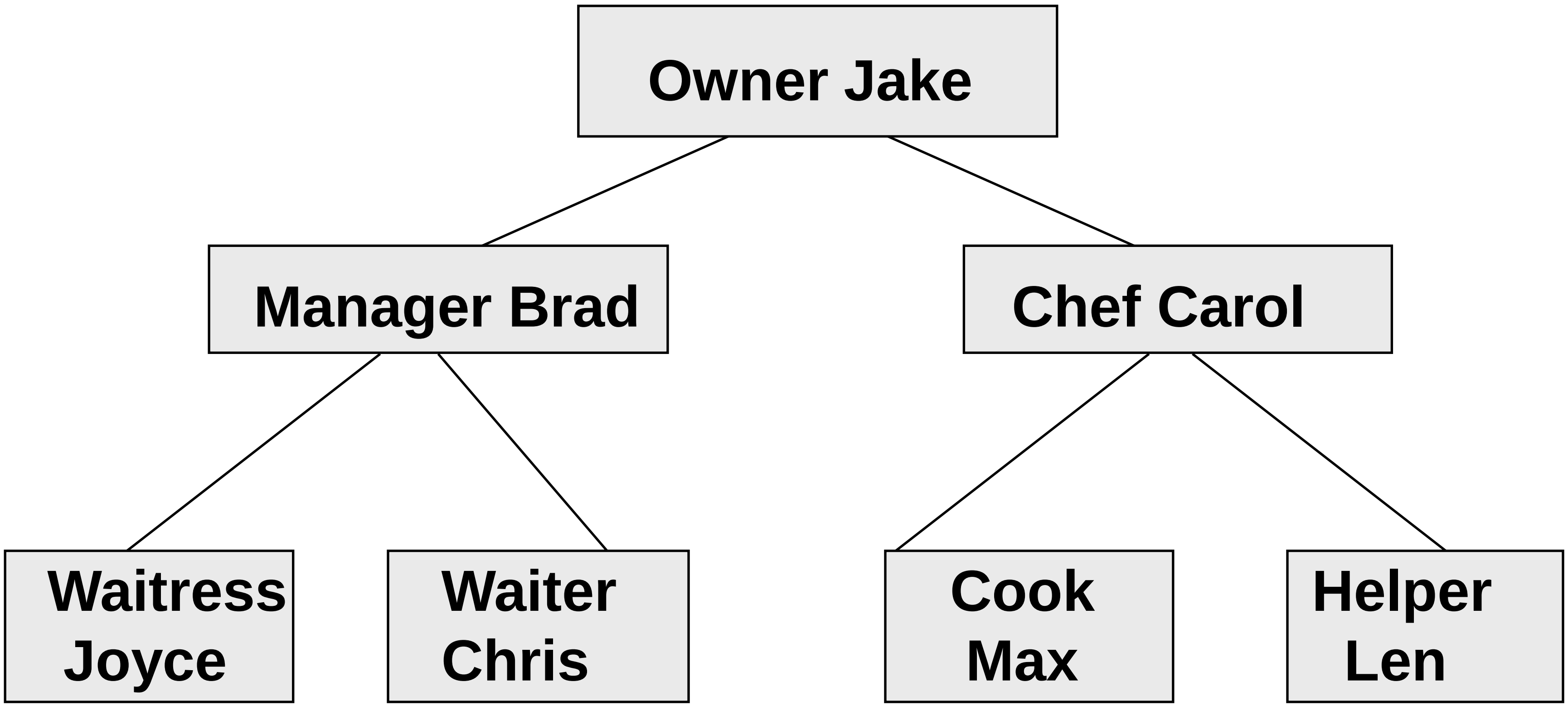
What if a different root is chosen?

Then, direct each edge away from the root.

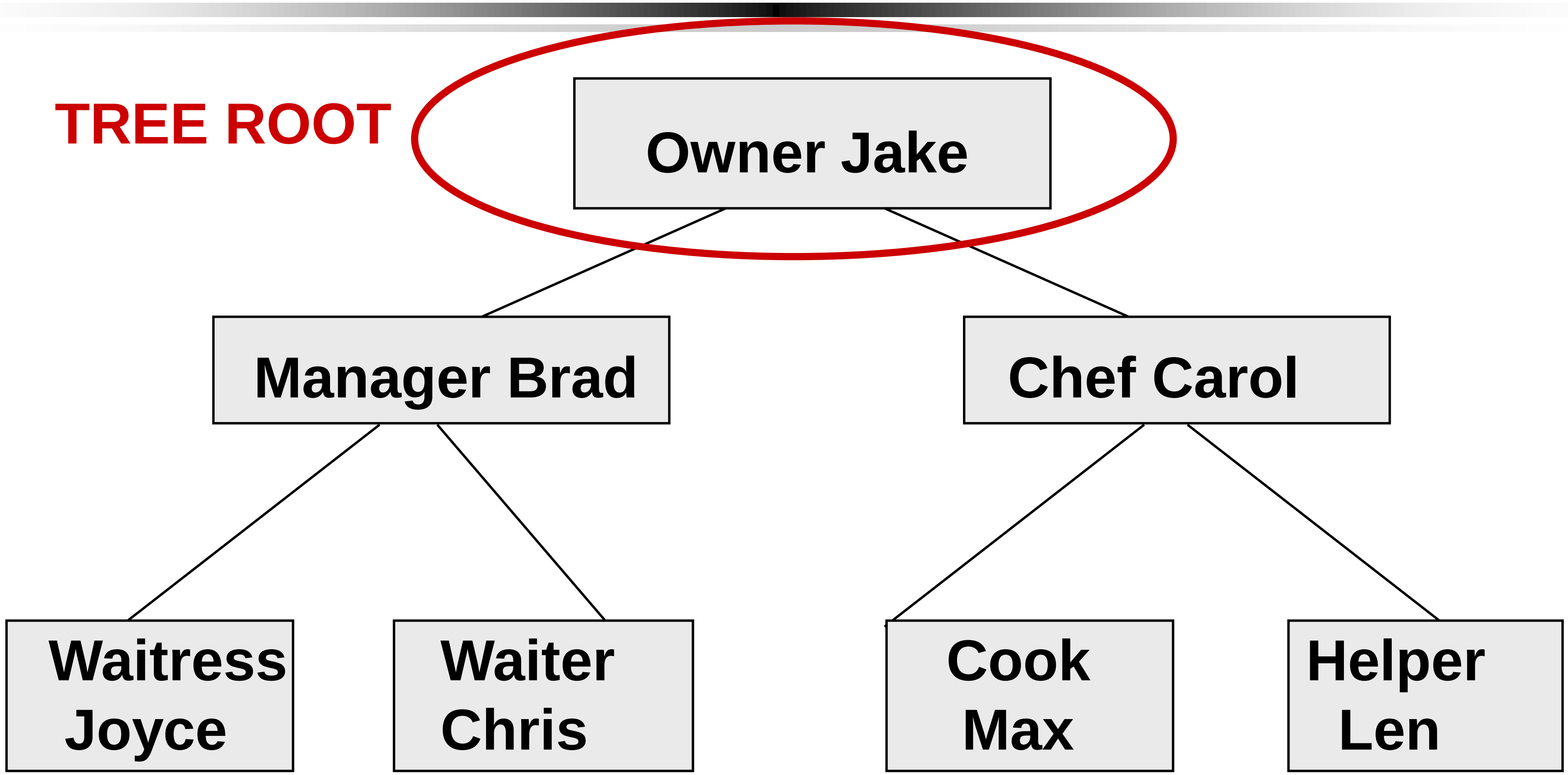


A different rooted tree results.

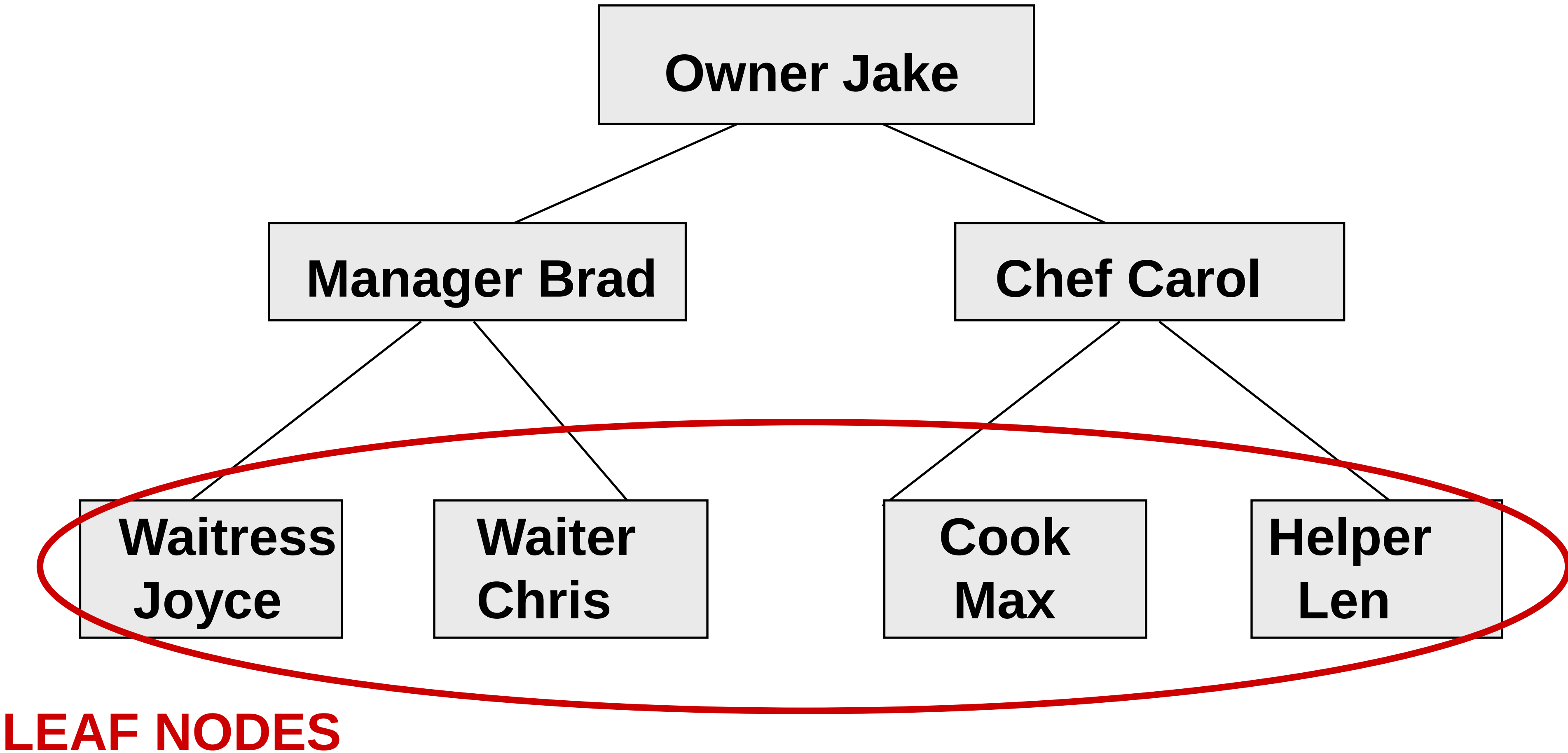
Jake's Pizza Shop Tree



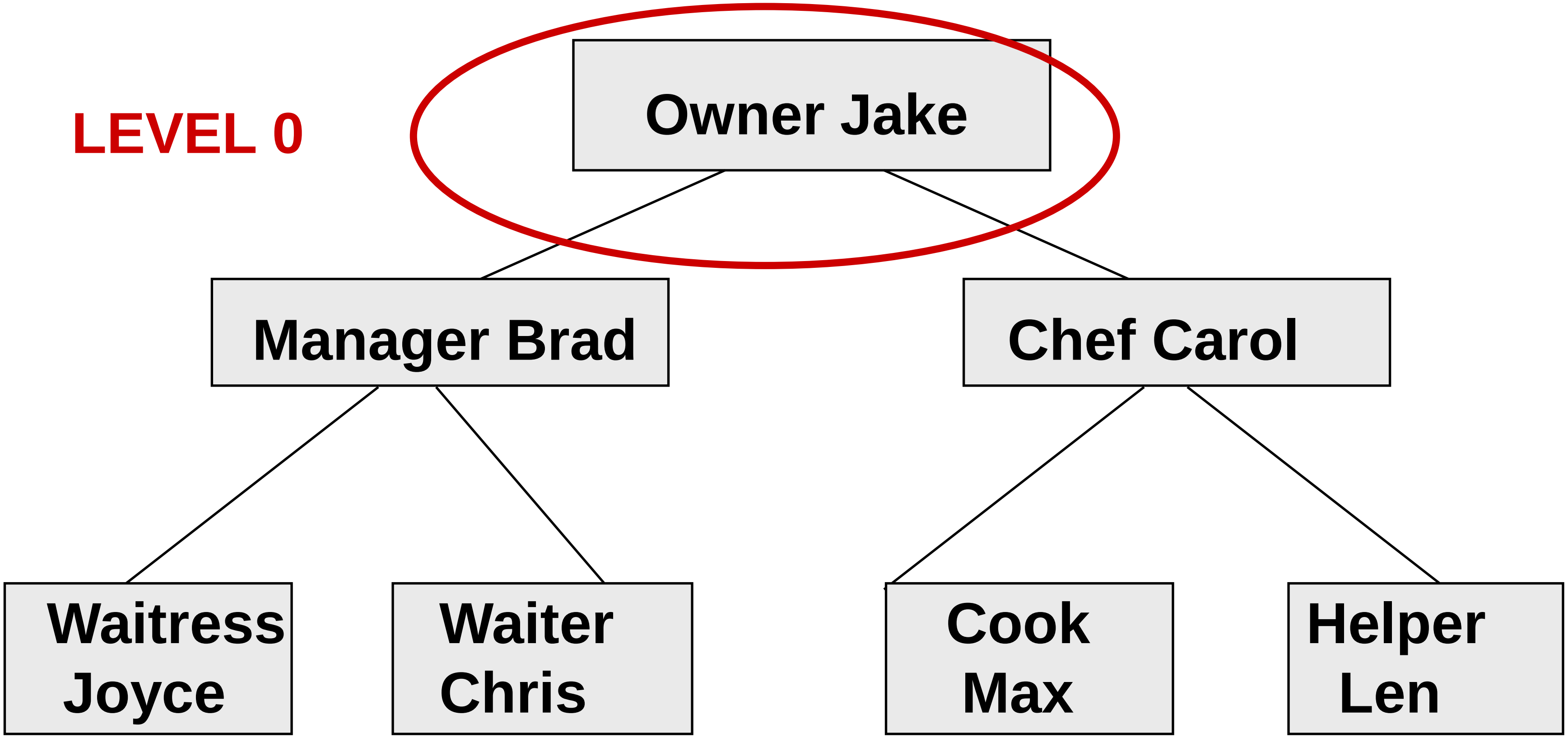
A Tree Has a Root



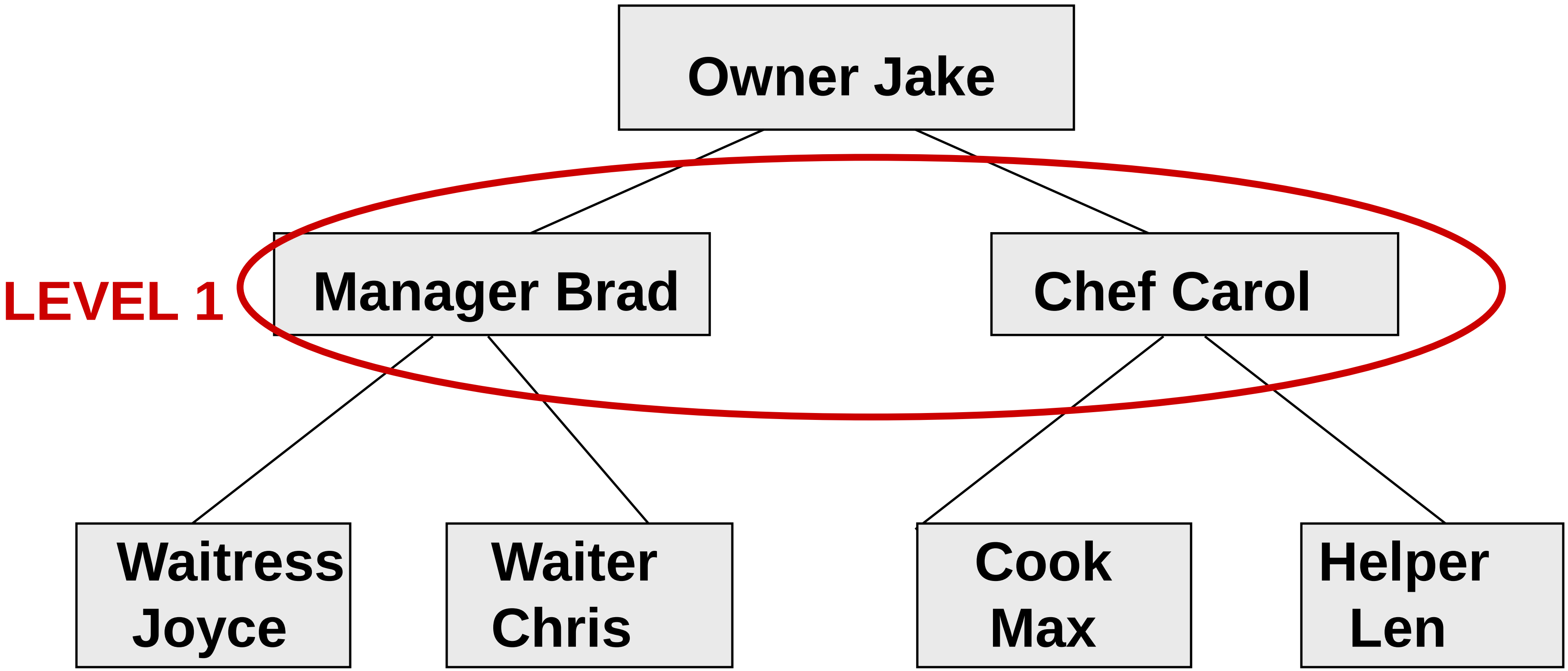
A Tree Has a Root



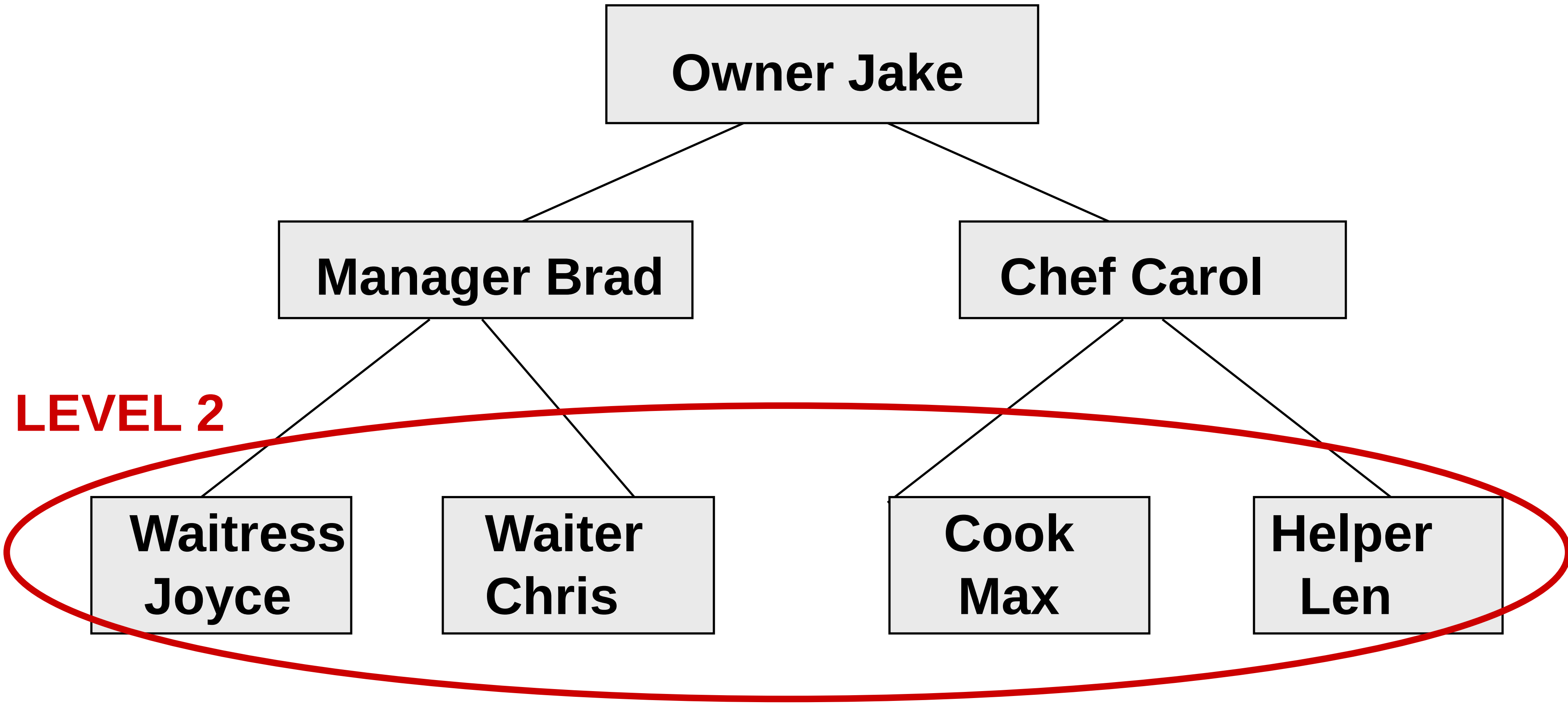
A Tree Has a Root



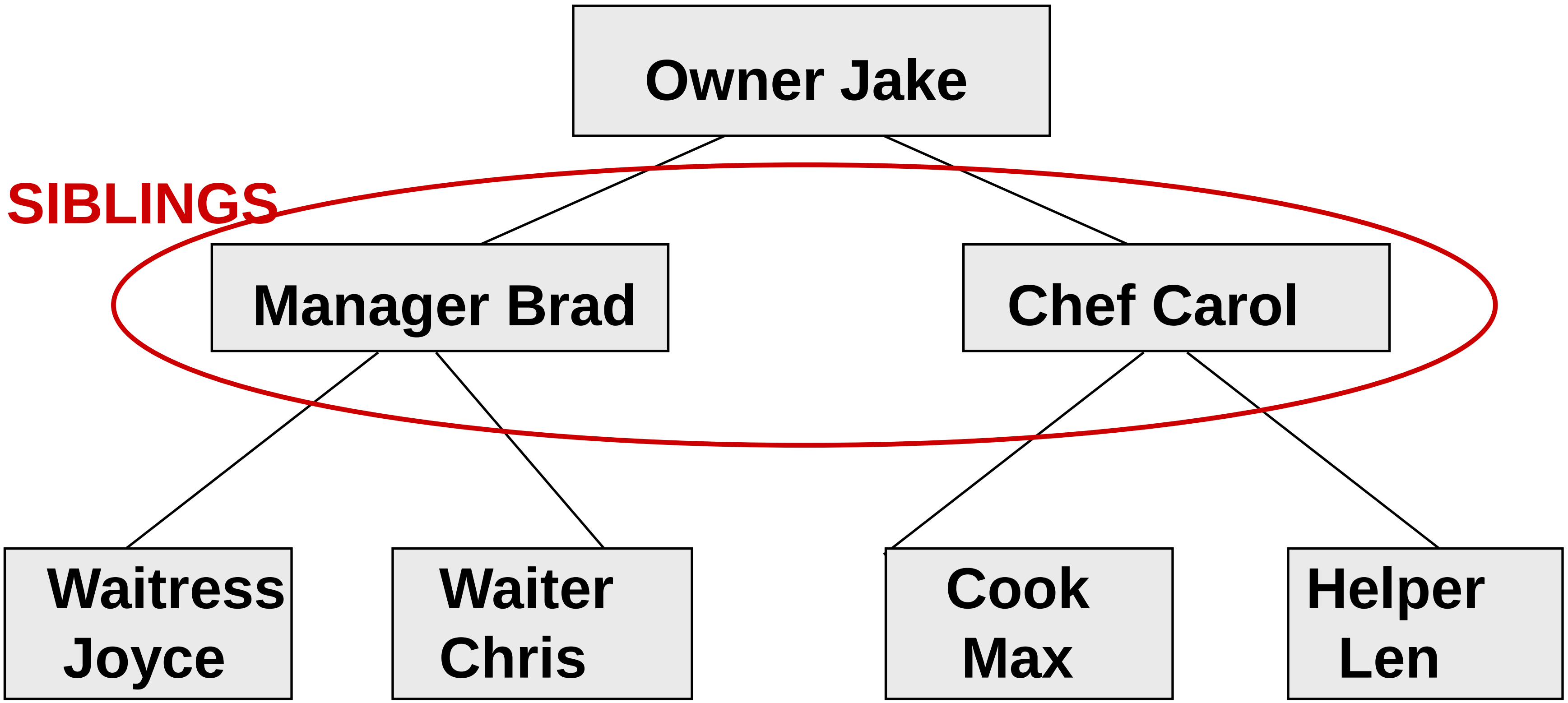
A Tree Has a Root



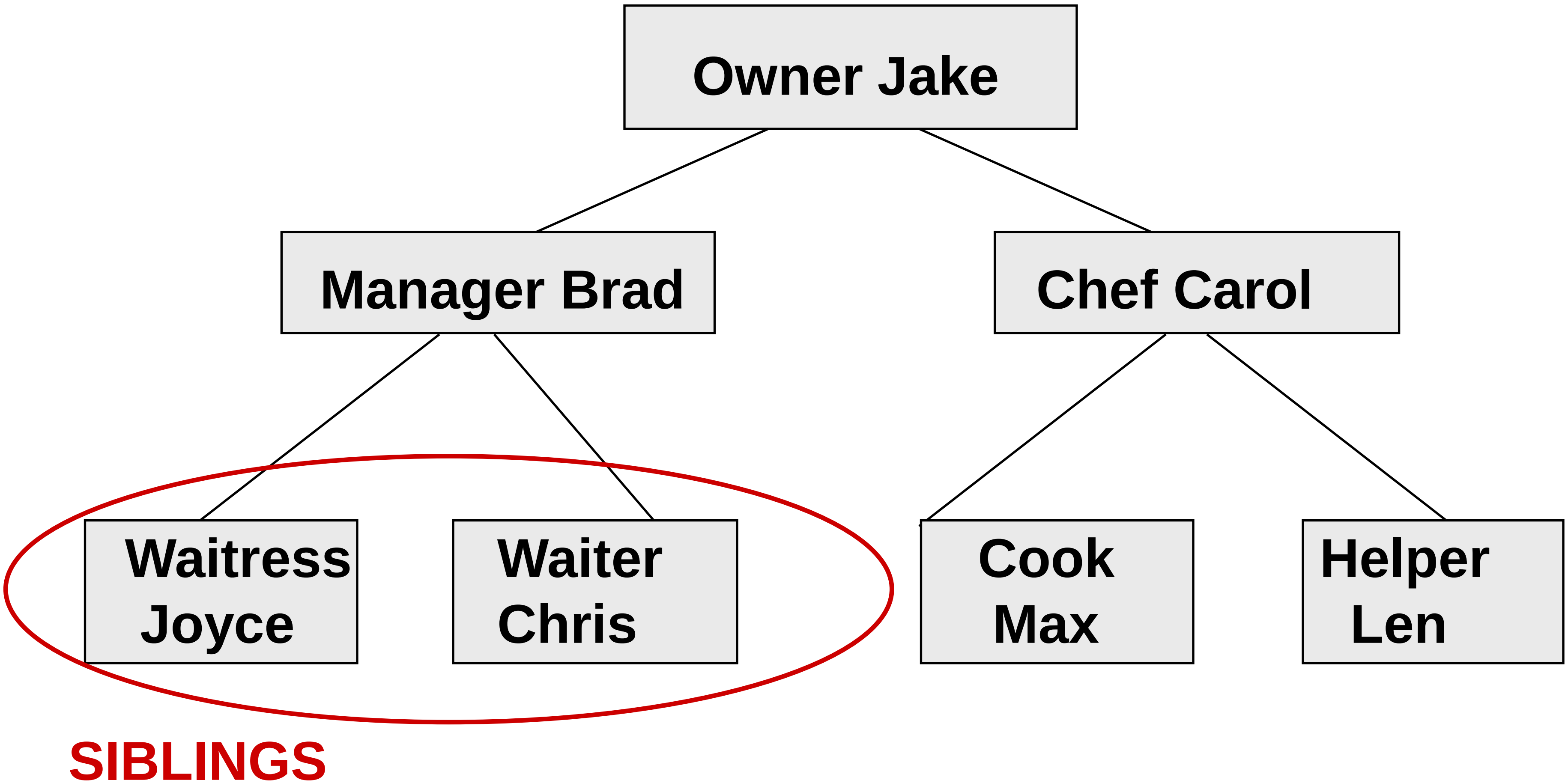
A Tree Has a Root



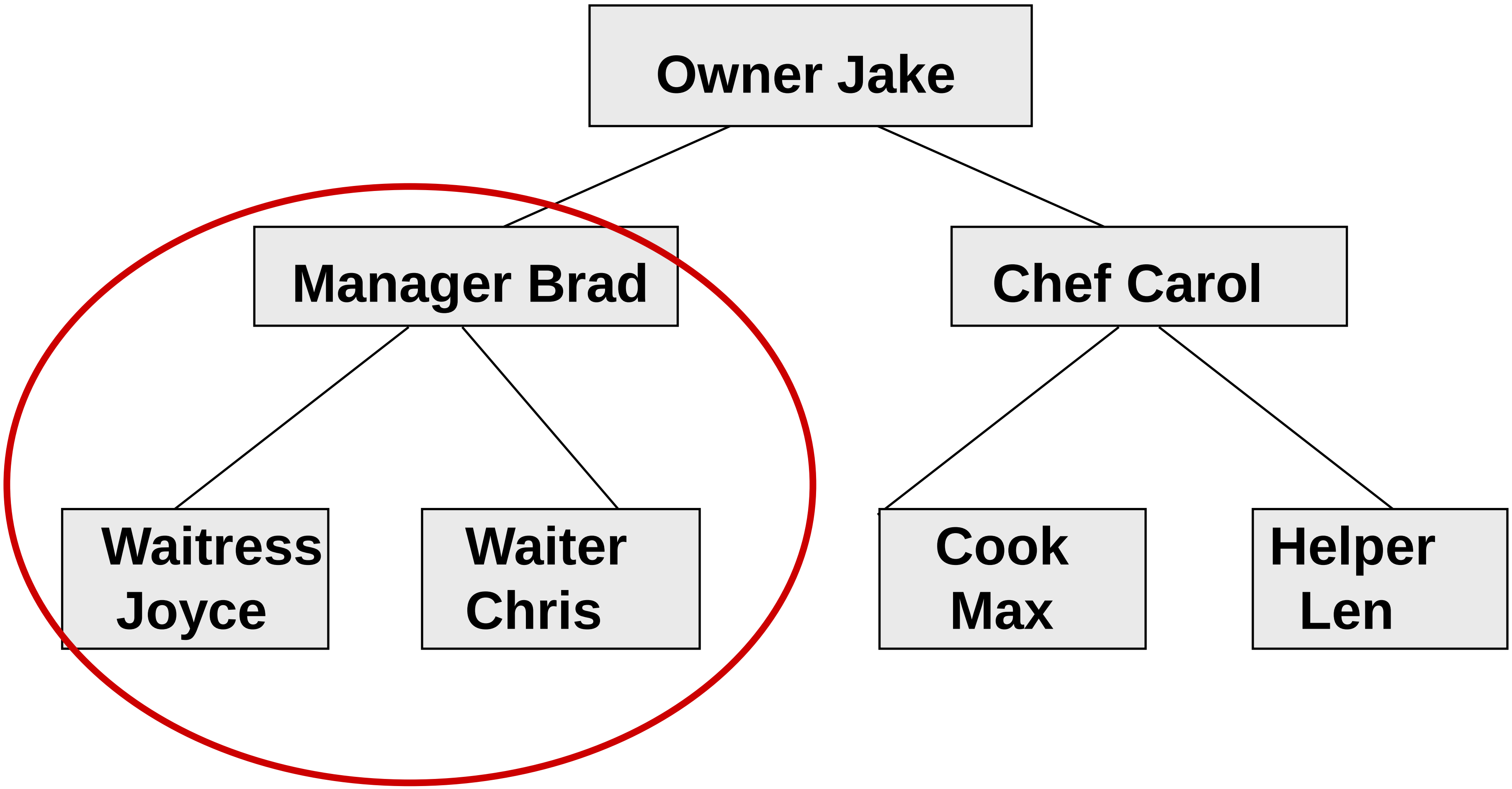
A Tree Has a Root



A Tree Has a Root

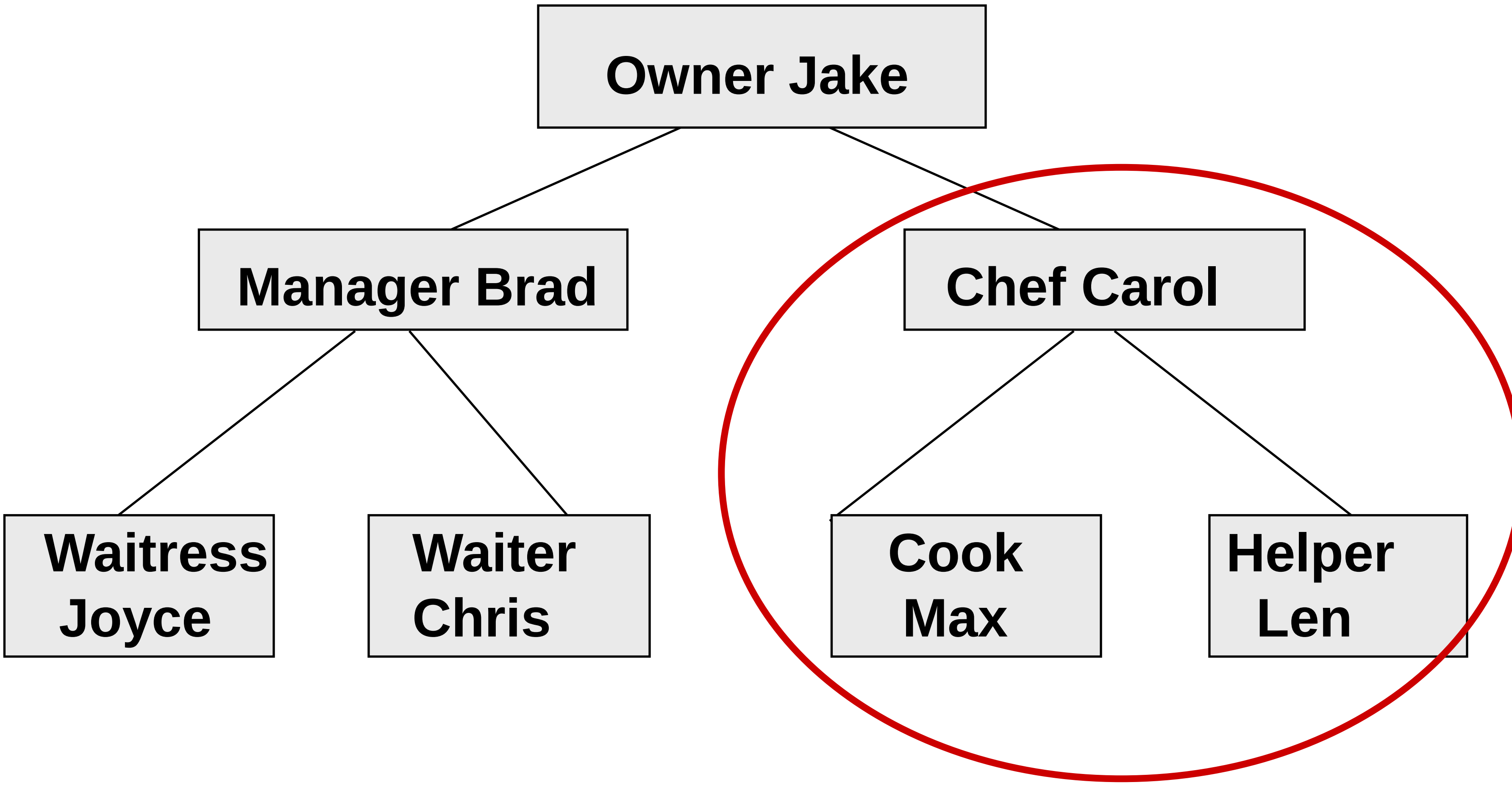


A Tree Has a Root



LEFT SUBTREE OF ROOT

A Tree Has a Root



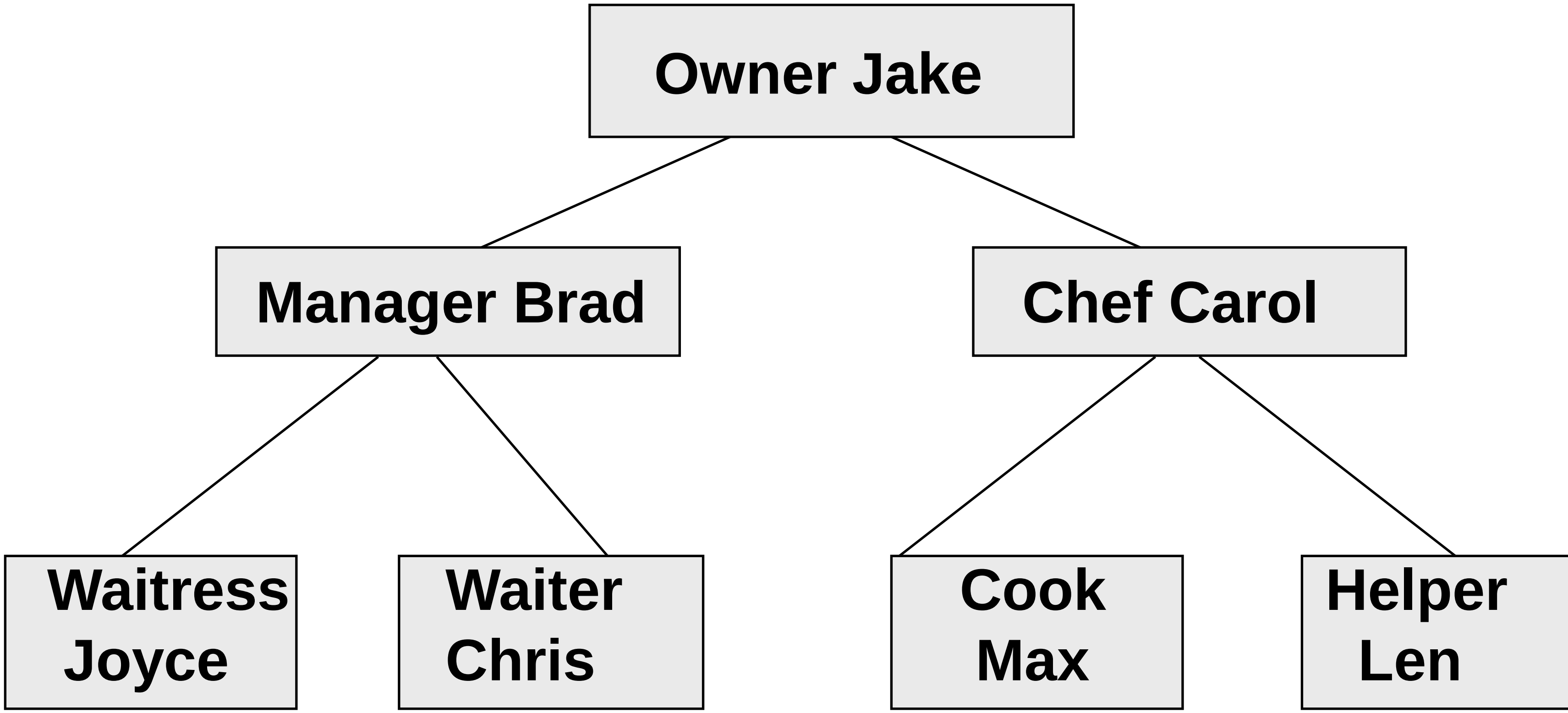
RIGHT SUBTREE OF ROOT

Internal Vertex

A vertex that has children is called an **internal vertex.**

The **subtree at vertex v** is the subgraph of the tree consisting of vertex v and its descendants and all edges incident to those descendants.

How many internal vertices?



Binary Tree

Definition 2′. A rooted tree is called a **binary tree** if every internal vertex has no more than 2 children.

The tree is called a **full** binary tree if every internal vertex has exactly 2 children.

Ordered Binary Tree

Definition 2′′. **An ordered rooted tree is a rooted tree where the children of each internal vertex are ordered.**

In an ordered binary tree, the two possible children of a vertex are called the left child and the right child, if they exist.

Tree Properties

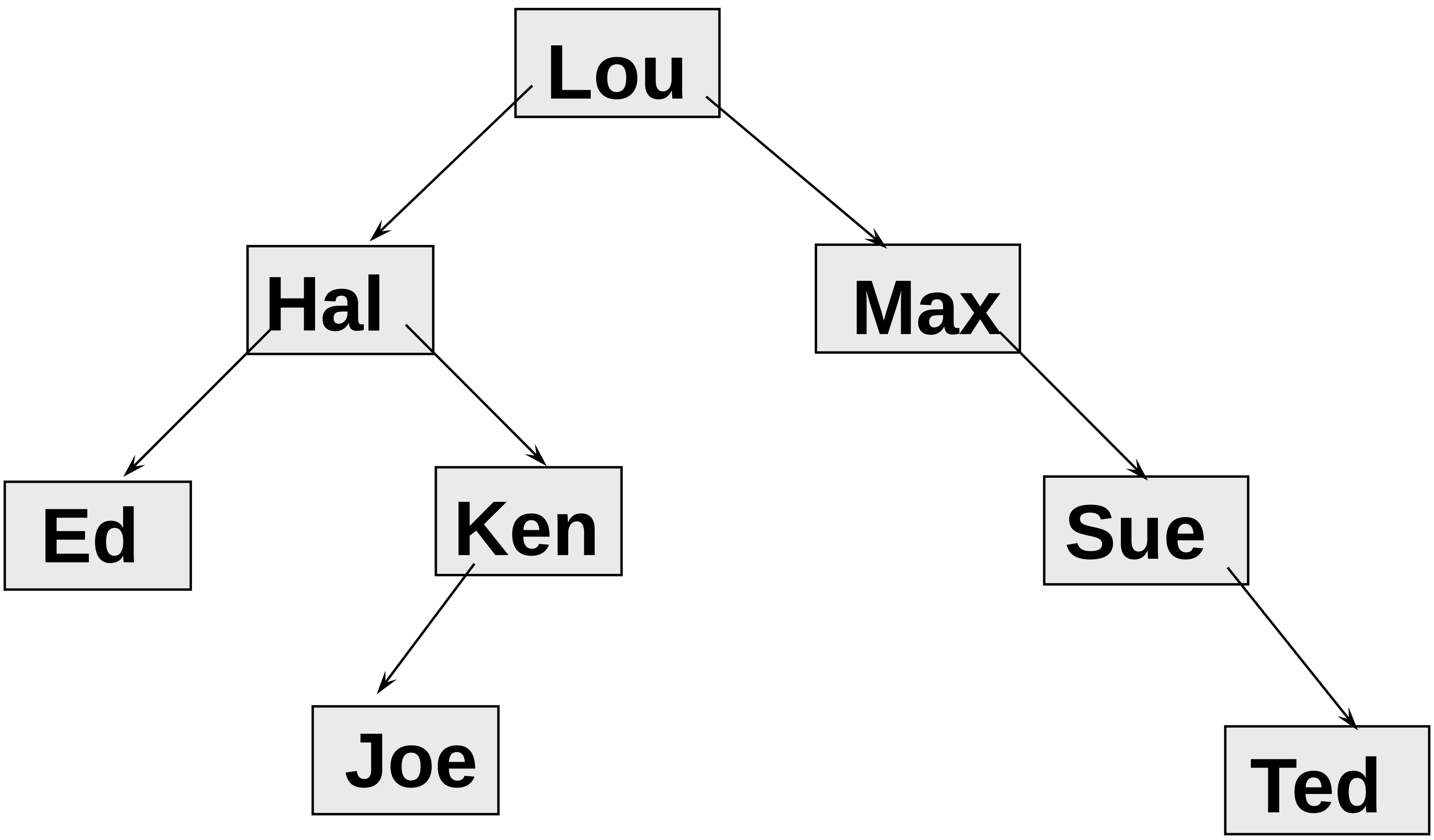
Theorem 2. A tree with N vertices has N-1 edges.

Theorem 5. There are at most 2^H leaves in a binary tree of height H.

Corallary. If a binary tree with L leaves is full and balanced, then its height is

$$H = \lceil \log_2 L \rceil.$$

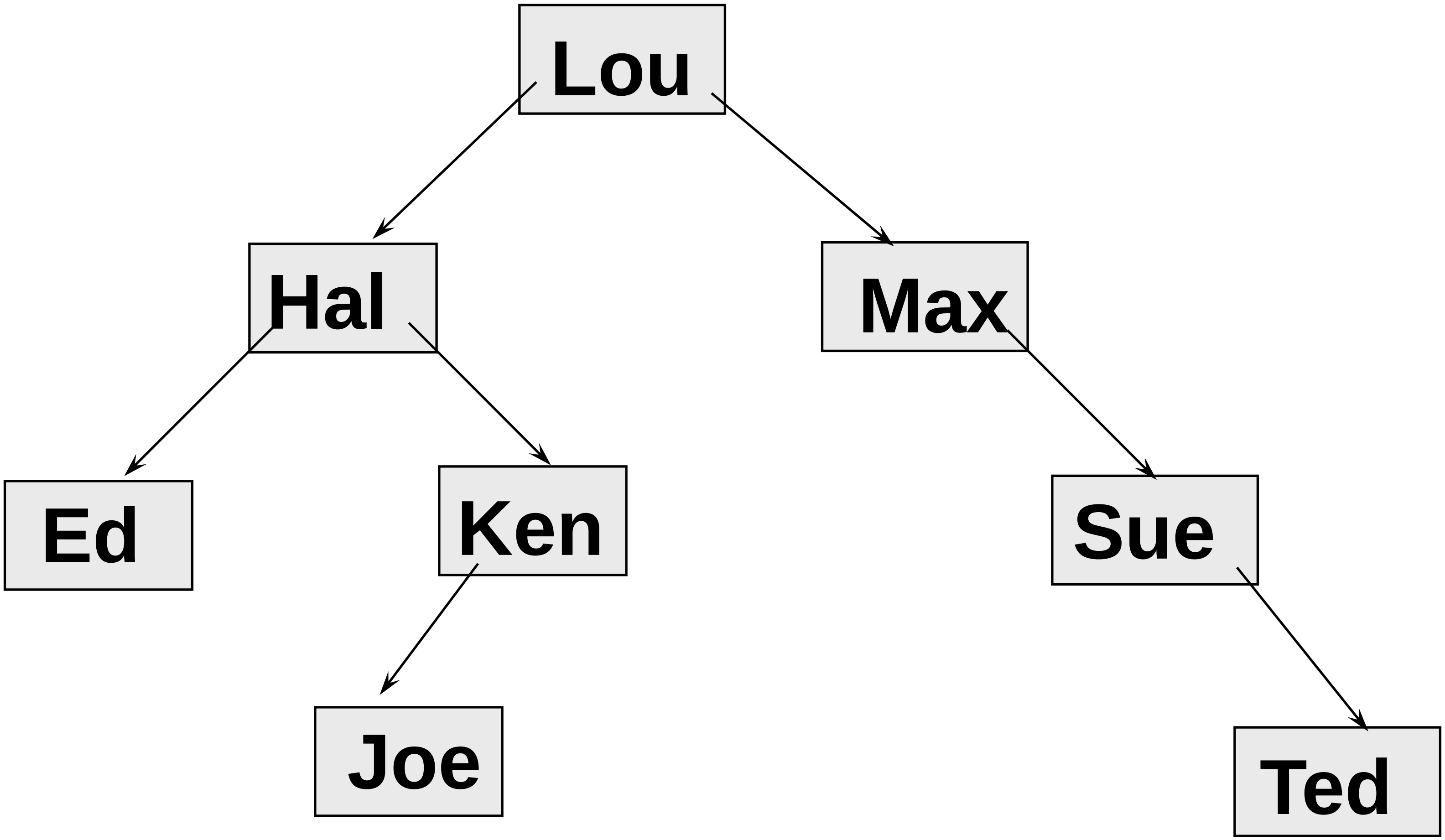
An Ordered Binary Tree



Parent

- The parent of a non-root vertex is the unique vertex u with a directed edge from u to v.

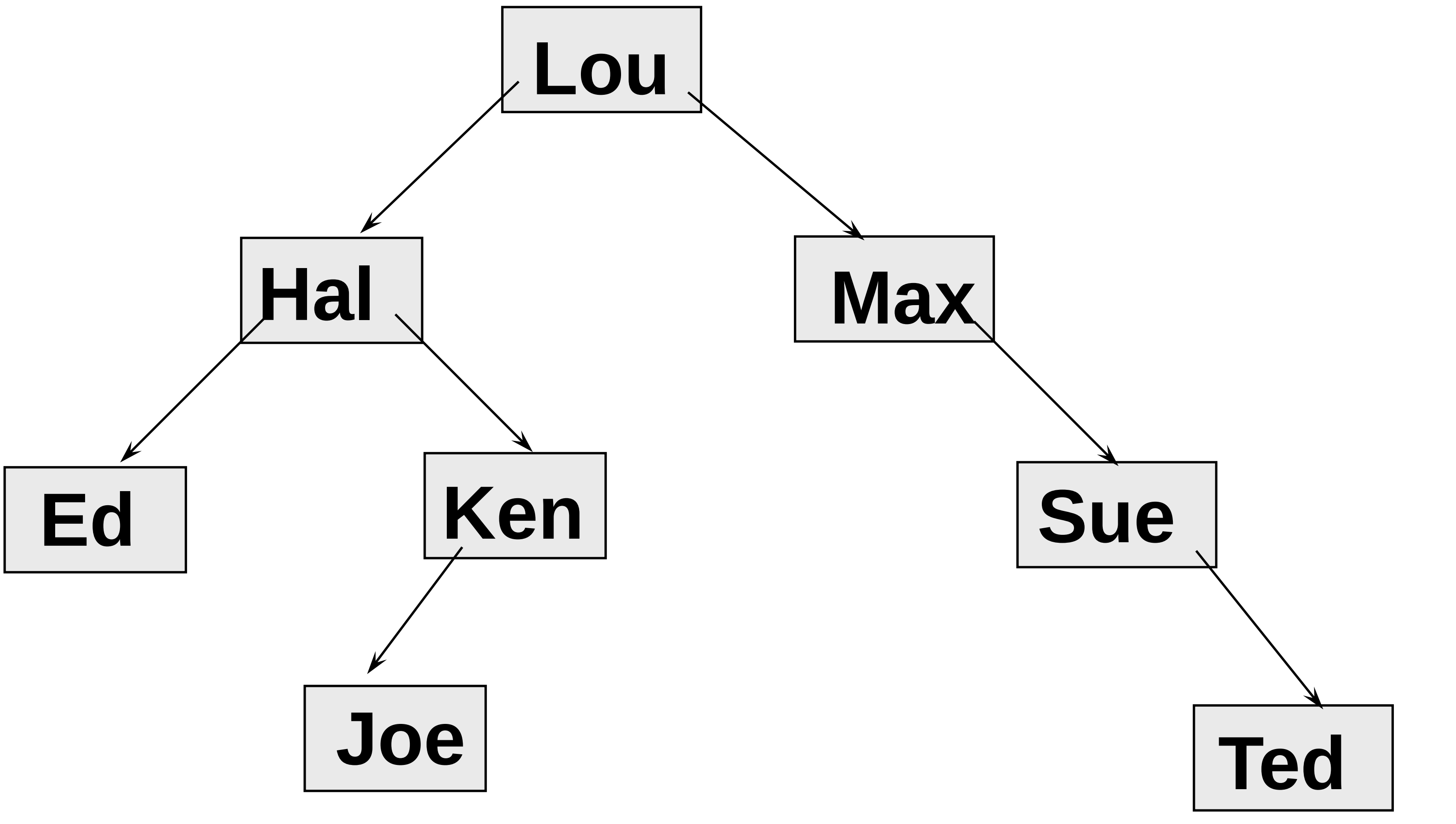
What is the parent of Ed?



Leaf

- **A vertex is called a leaf if it has no children.**

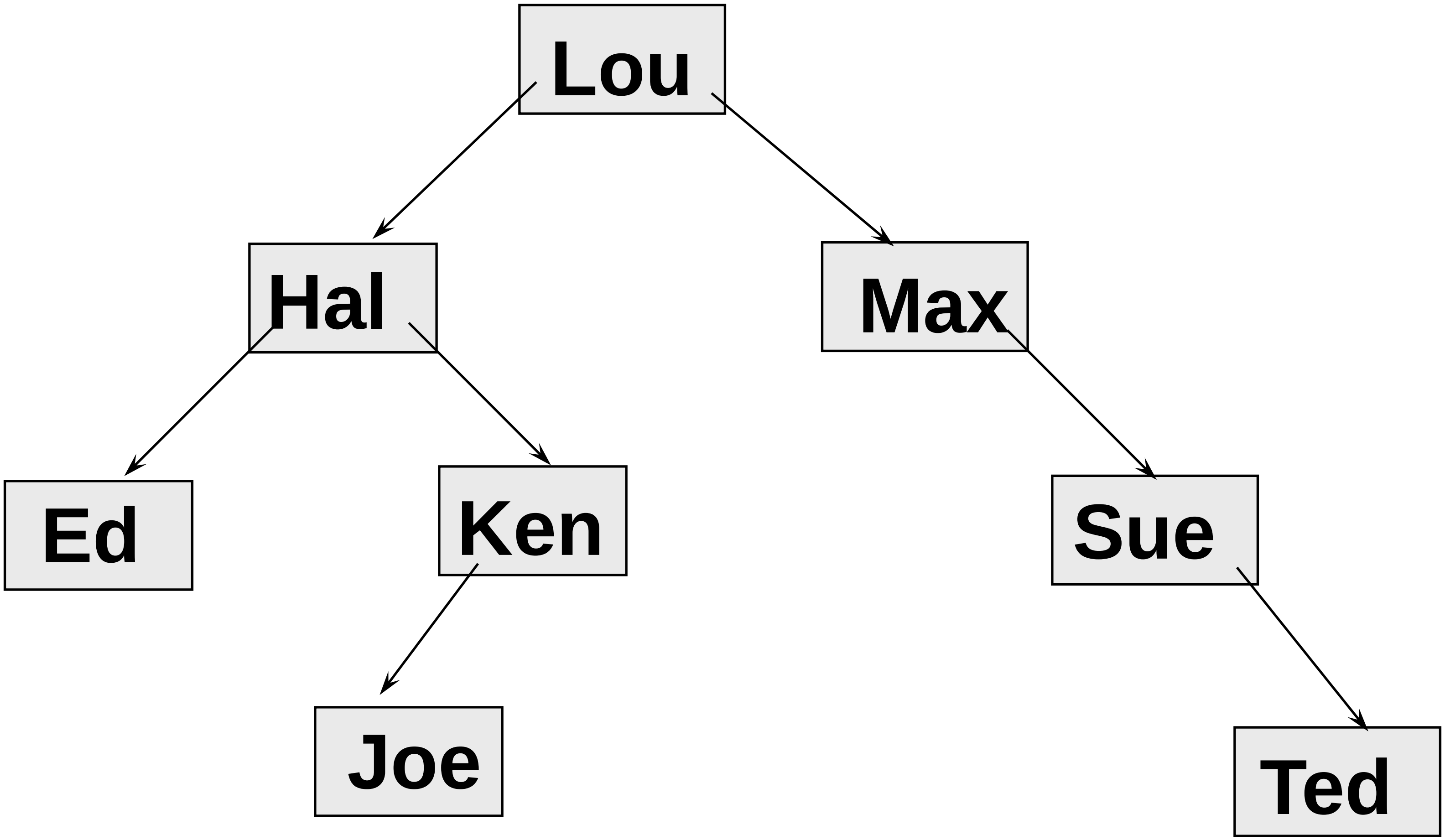
How many leaves?



Ancestors

- The **ancestors of a non-root vertex** are all the vertices in the path from root to this vertex.

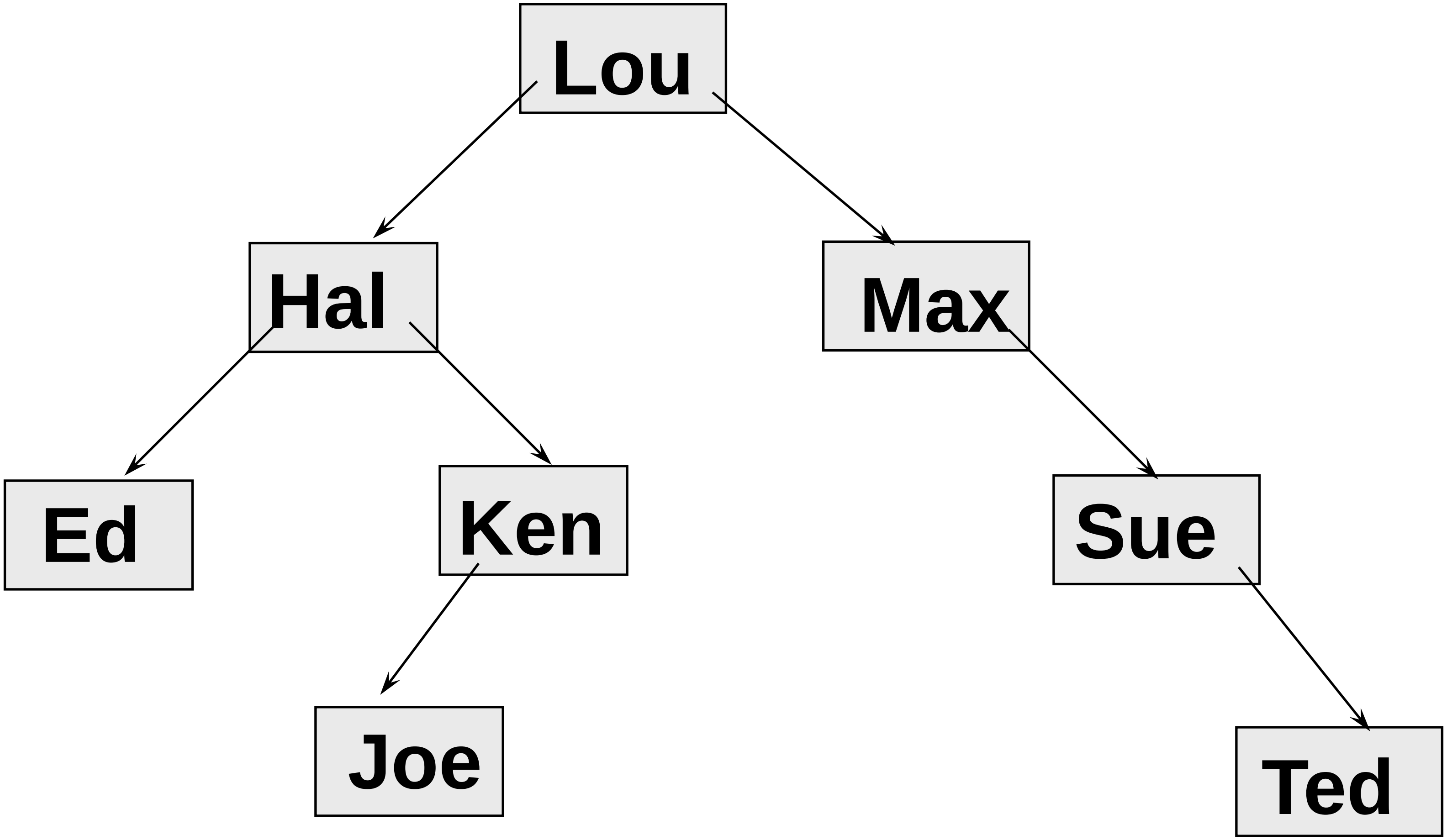
How many ancestors of Ken?



Descendants

- **The descendants of vertex v are all the vertices that have v as an ancestor.**

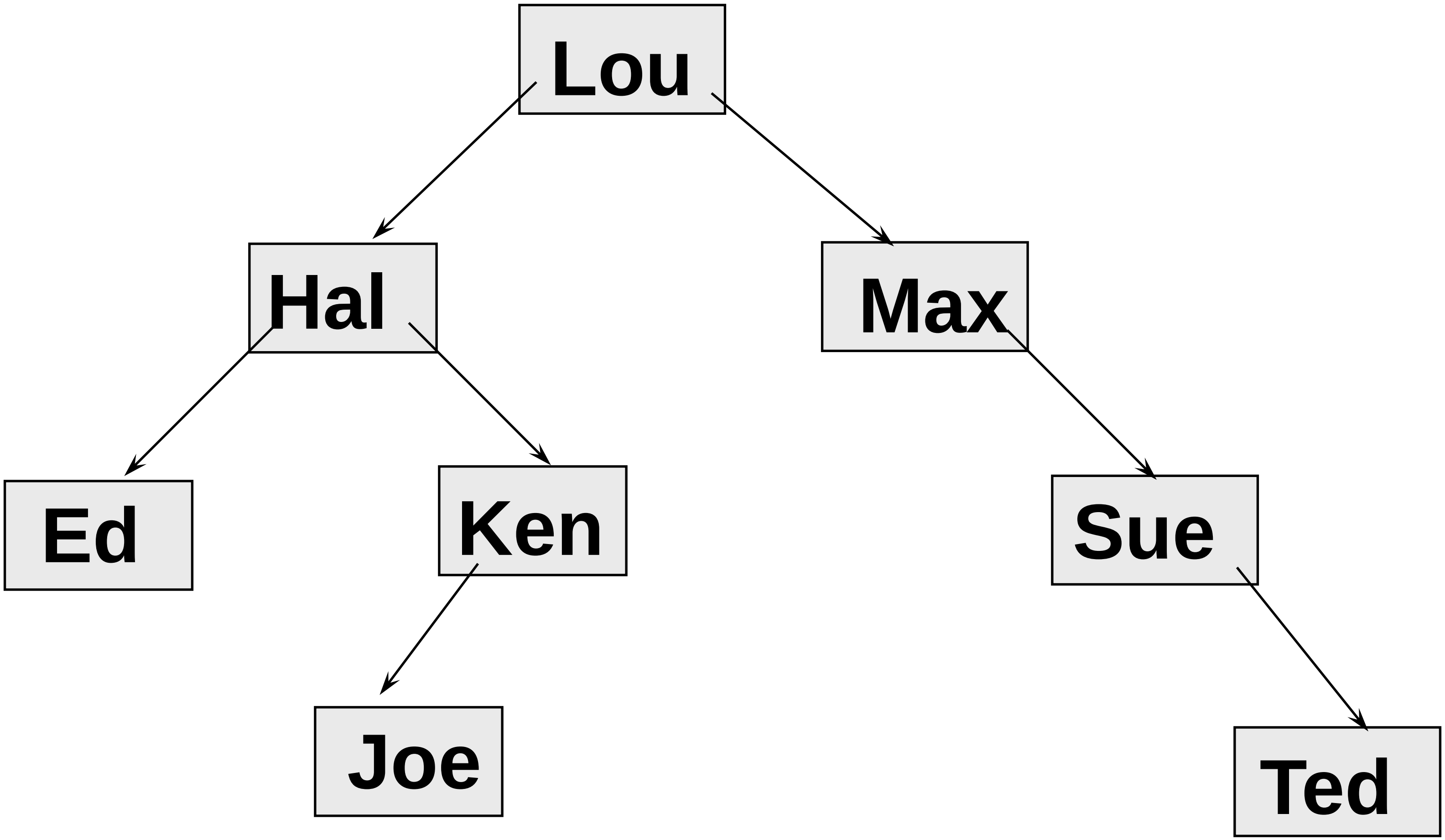
How many descendants of Hal?



Level

- **The level of vertex v in a rooted tree is the length of the unique path from the root to v.**

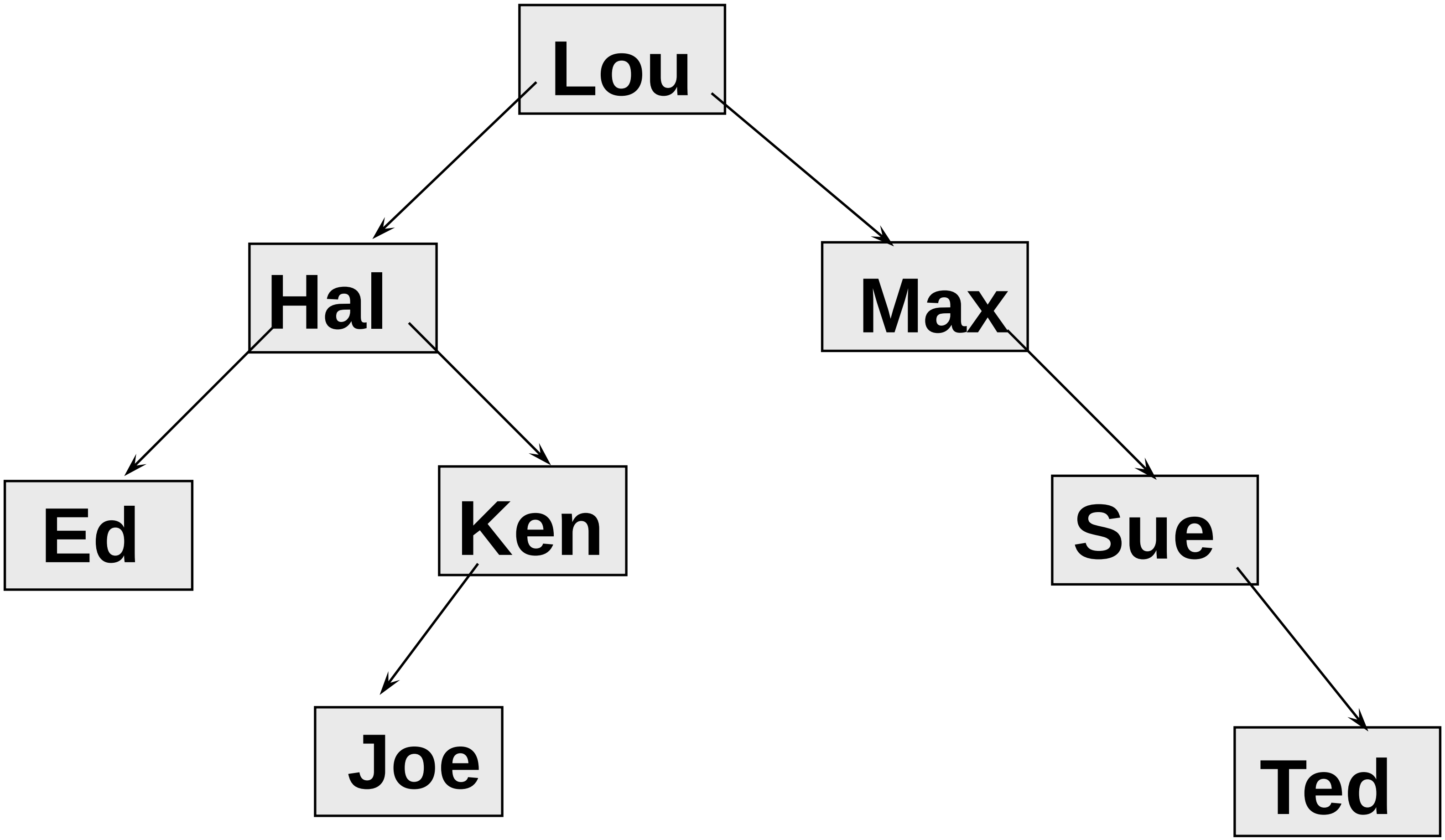
What is the level of Ted?



Height

- The height of a rooted tree is the maximum of the levels of its vertices.

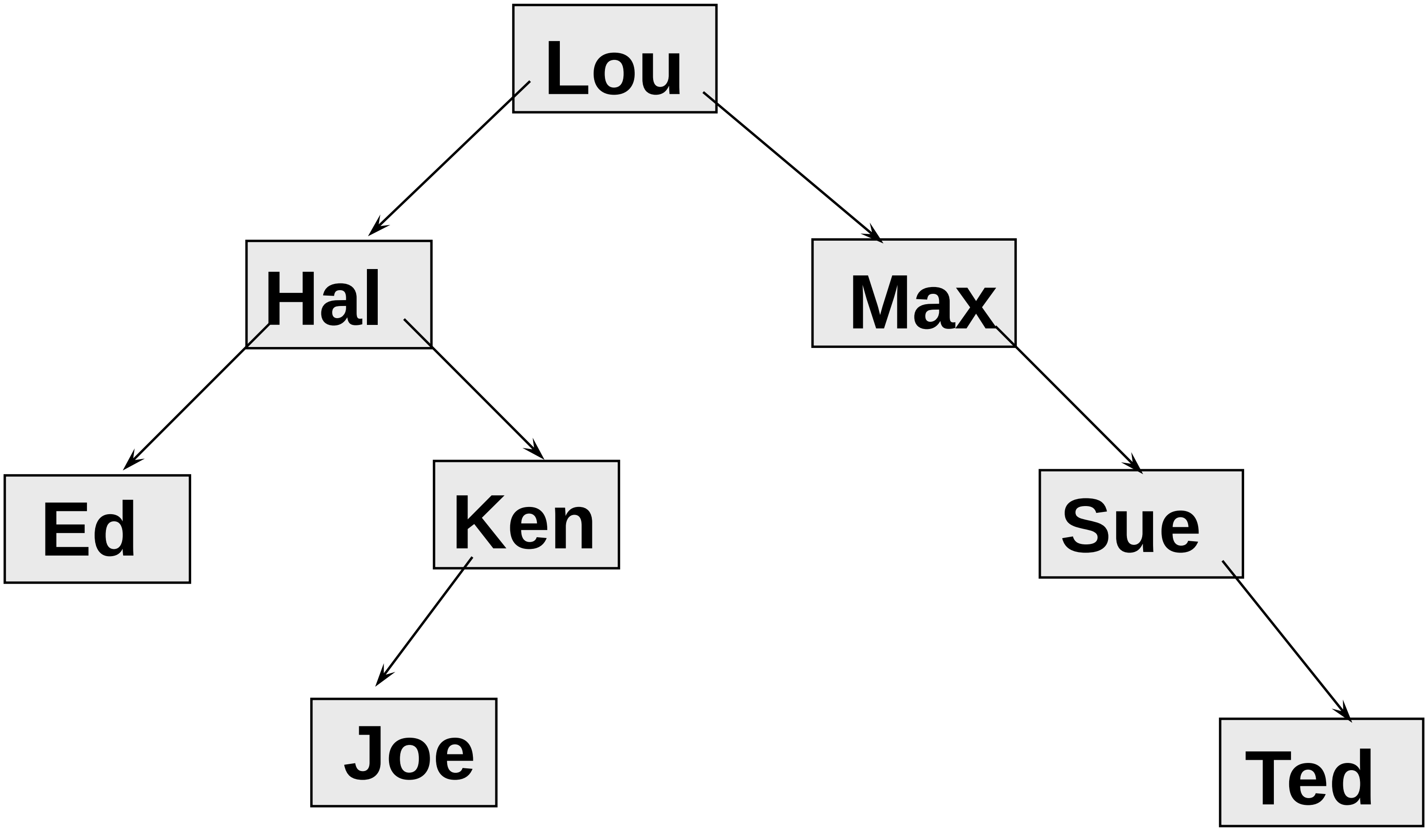
What is the height?



Balanced

- A rooted binary tree of height H is called balanced if all its leaves are at levels H or H-1.

Is this binary tree balanced?



Searching takes time . . .

So the goal in computer programs is to find any stored item efficiently when all stored items are ordered.

A Binary Search Tree can be used to store items in its vertices. It enables efficient searches.

A Binary Search Tree (BST) is . . .

A special kind of binary tree in which:

1. **Each vertex contains a distinct key value,**
2. **The key values in the tree can be compared using "greater than" and "less than", and**
3. **The key value of each vertex in the tree is less than every key value in its right subtree, and greater than every key value in its left subtree.**

Shape of a binary search tree . . .

Depends on its key values and their order of insertion.

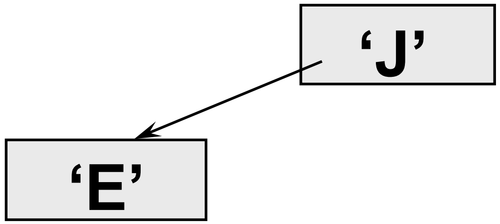
Insert the elements ‘J’ ‘E’ ‘F’ ‘T’ ‘A’ in that order.

The first value to be inserted is put into the root.

$\boxed{\text{‘J’}}$

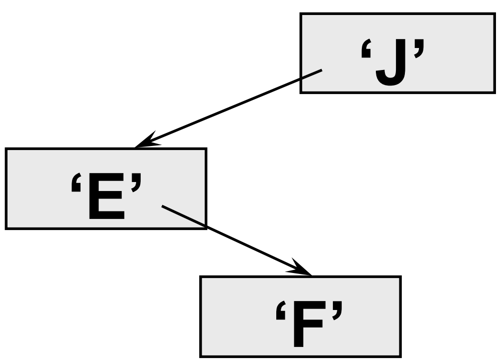
Inserting ‘E’ into the BST

Thereafter, each value to be inserted begins by comparing itself to the value in the root, moving left it is less, or moving right if it is greater. This continues at each level until it can be inserted as a new leaf.



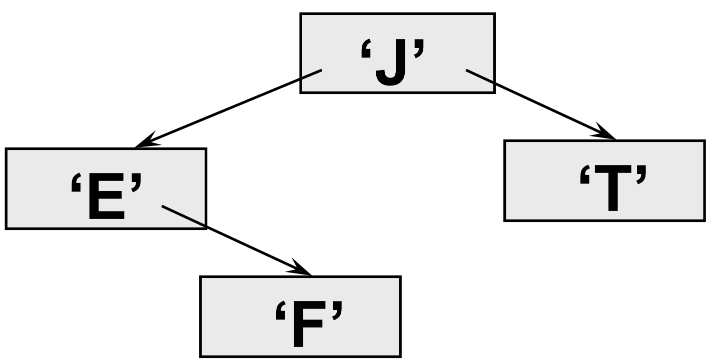
Inserting ‘F’ into the BST

Begin by comparing ‘F’ to the value in the root, moving left it is less, or moving right if it is greater. This continues until it can be inserted as a leaf.



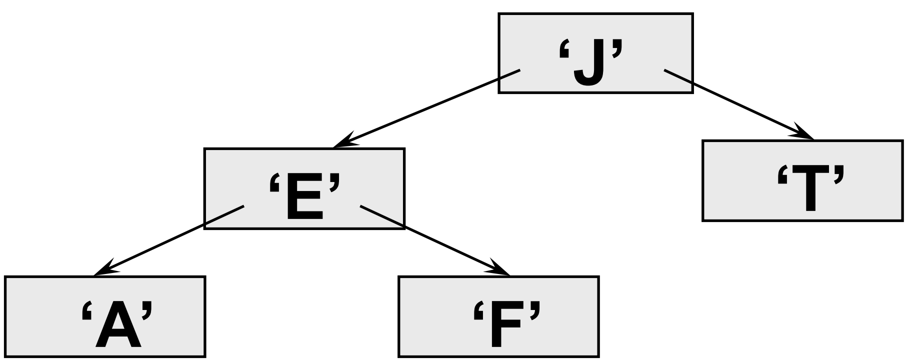
Inserting ‘T’ into the BST

Begin by comparing ‘T’ to the value in the root,
moving left it is less, or moving right if it is greater.
This continues until it can be inserted as a leaf.



Inserting ‘A’ into the BST

Begin by comparing ‘A’ to the value in the root, moving left it is less, or moving right if it is greater. This continues until it can be inserted as a leaf.



What binary search tree . . .

is obtained by inserting

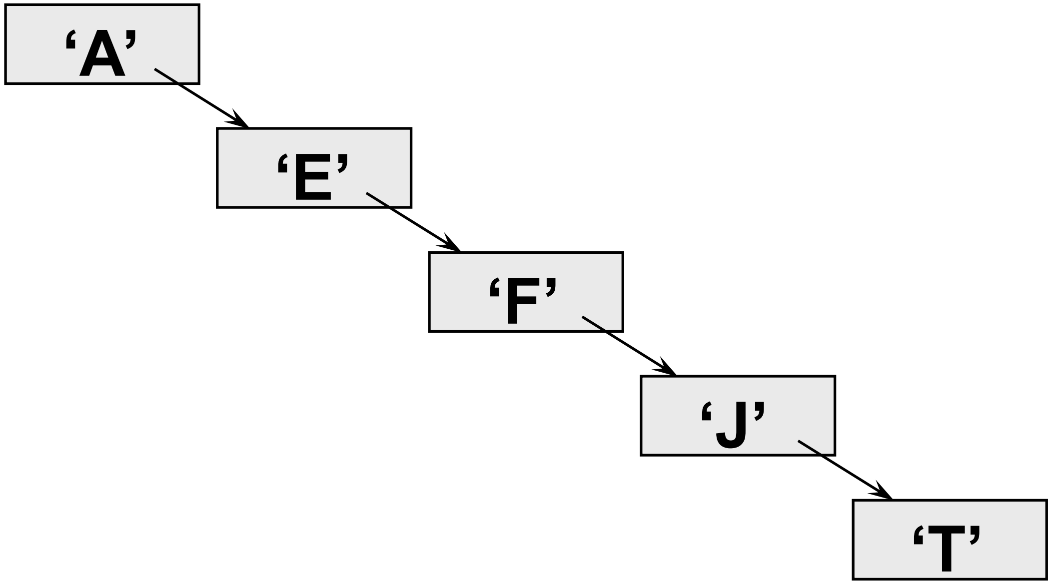
the elements ‘A’ ‘E’ ‘F’ ‘J’ ‘T’ in that order?

‘A’

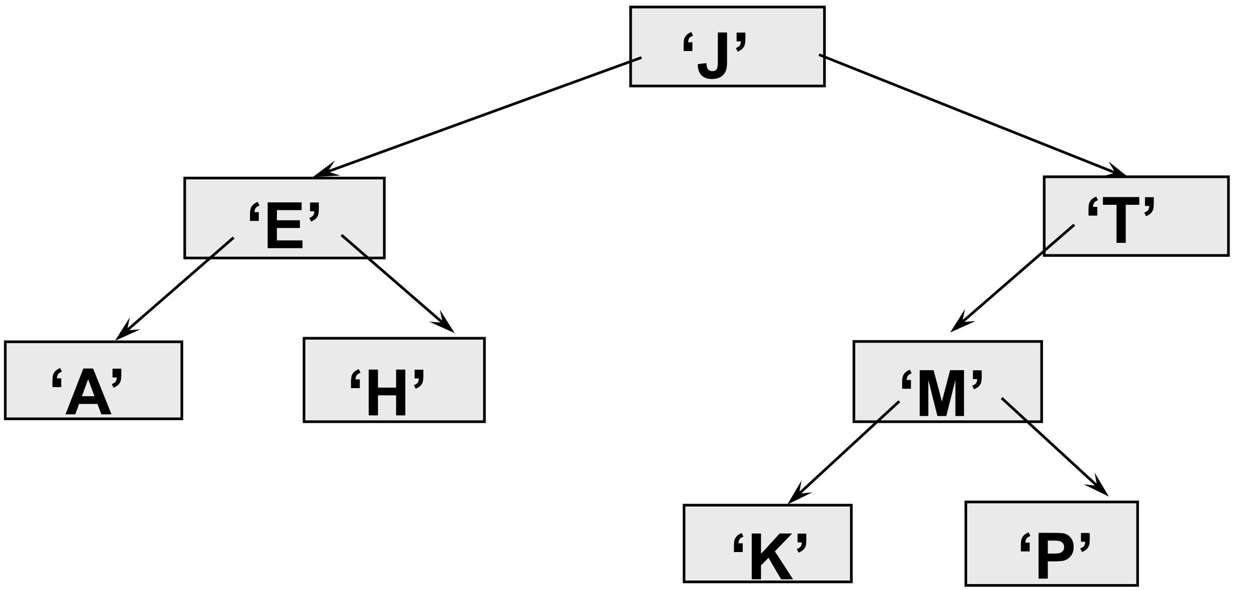
Binary search tree . . .

obtained by inserting

the elements 'A' 'E' 'F' 'J' 'T' in that order.



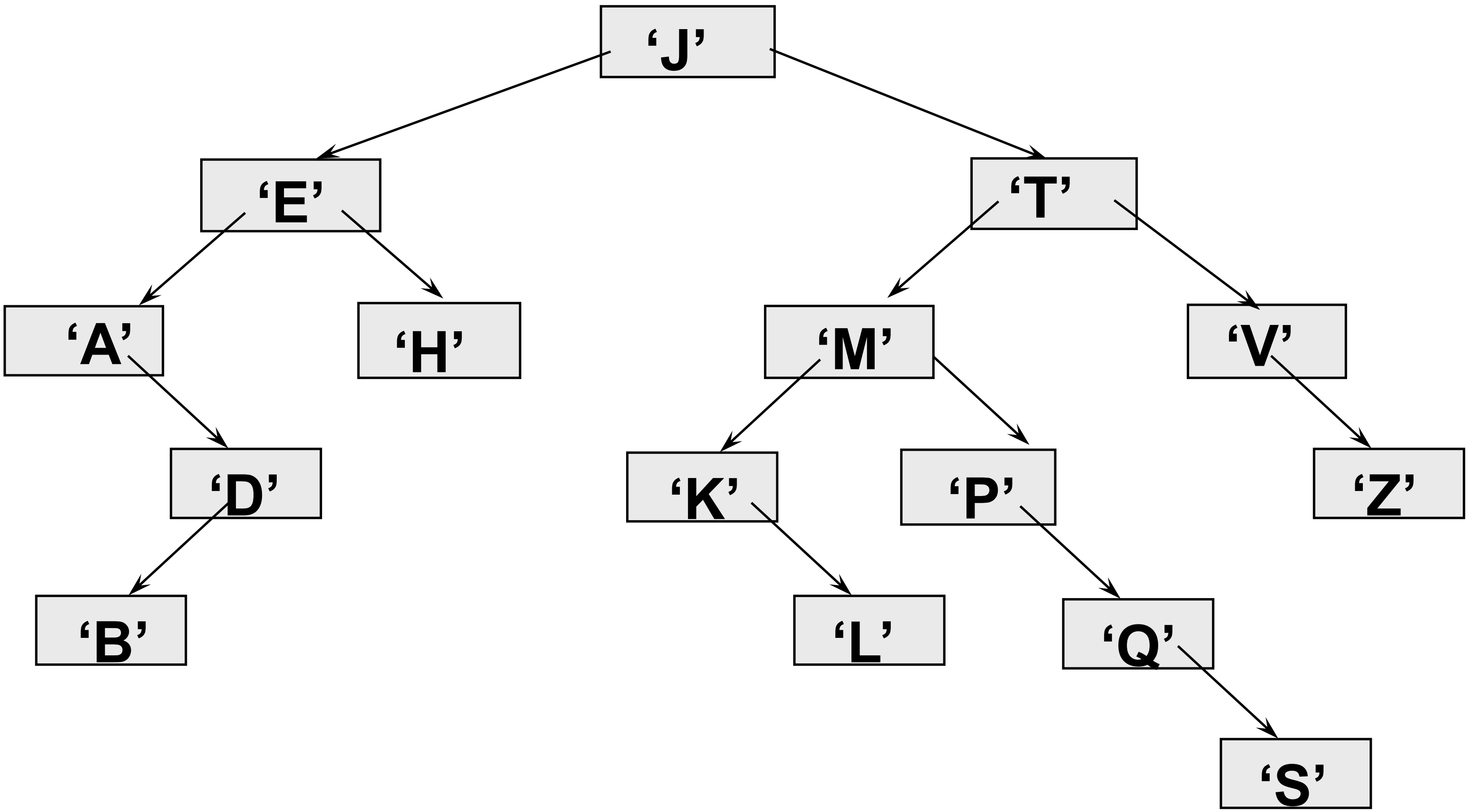
Another binary search tree



Add nodes containing these values in this order:

‘D’ ‘B’ ‘L’ ‘Q’ ‘S’ ‘V’ ‘Z’

Is ‘F’ in the binary search tree?



Traversal Algorithms

- A traversal algorithm is a procedure for systematically visiting every vertex of an ordered binary tree.
- Tree traversals are defined recursively.
- Three traversals are named:

  preorder,

  inorder,

  postorder.

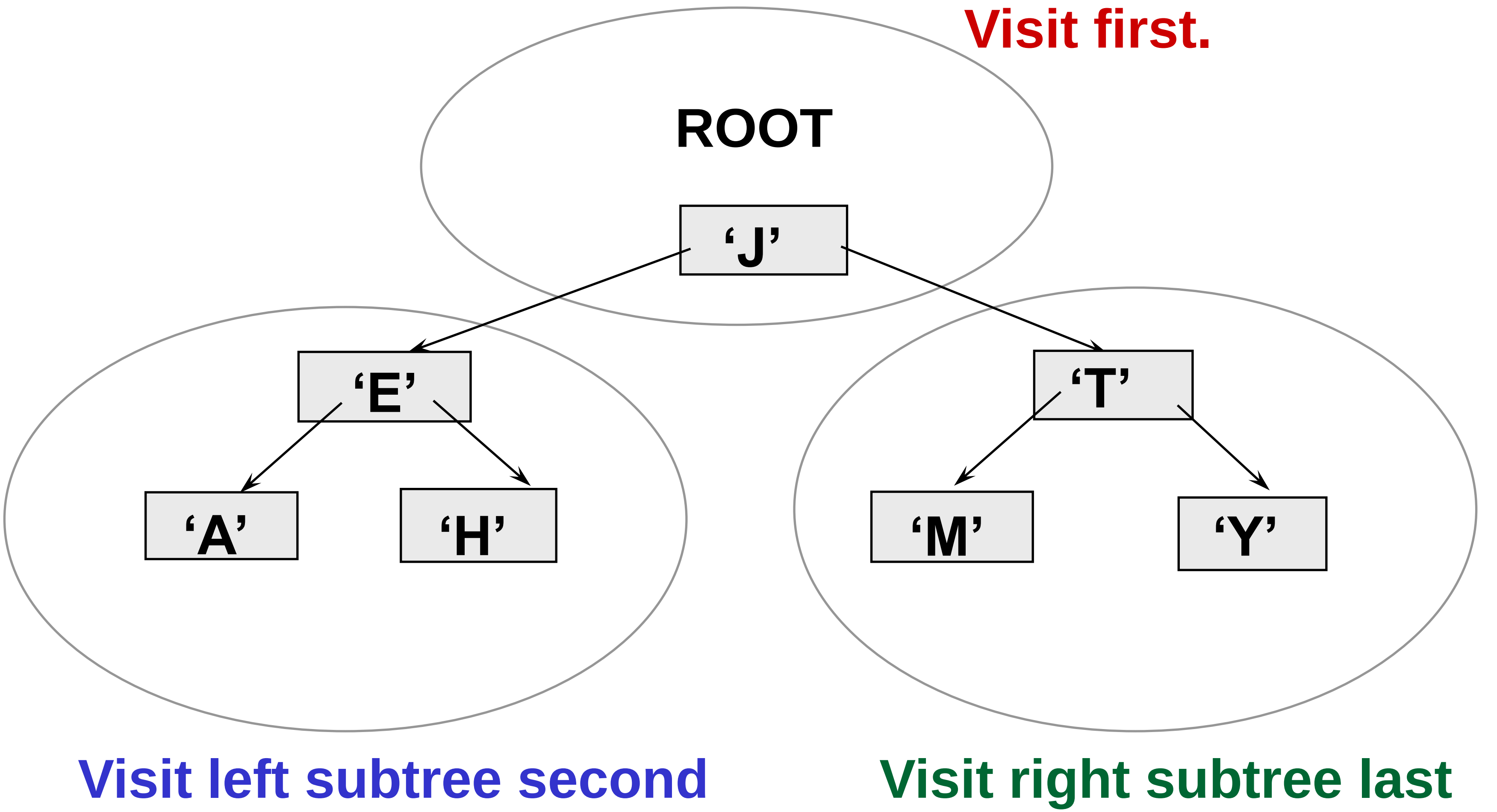
PREORDER Traversal Algorithm

Let T be an ordered binary tree with root r.

If T has only r, then r is the preorder traversal.

Otherwise, suppose T_1, T_2 are the left and right subtrees at r. The preorder traversal begins by visiting r. Then traverses T_1 in preorder, then traverses T_2 in preorder.

Preorder Traversal: J E A H T M Y



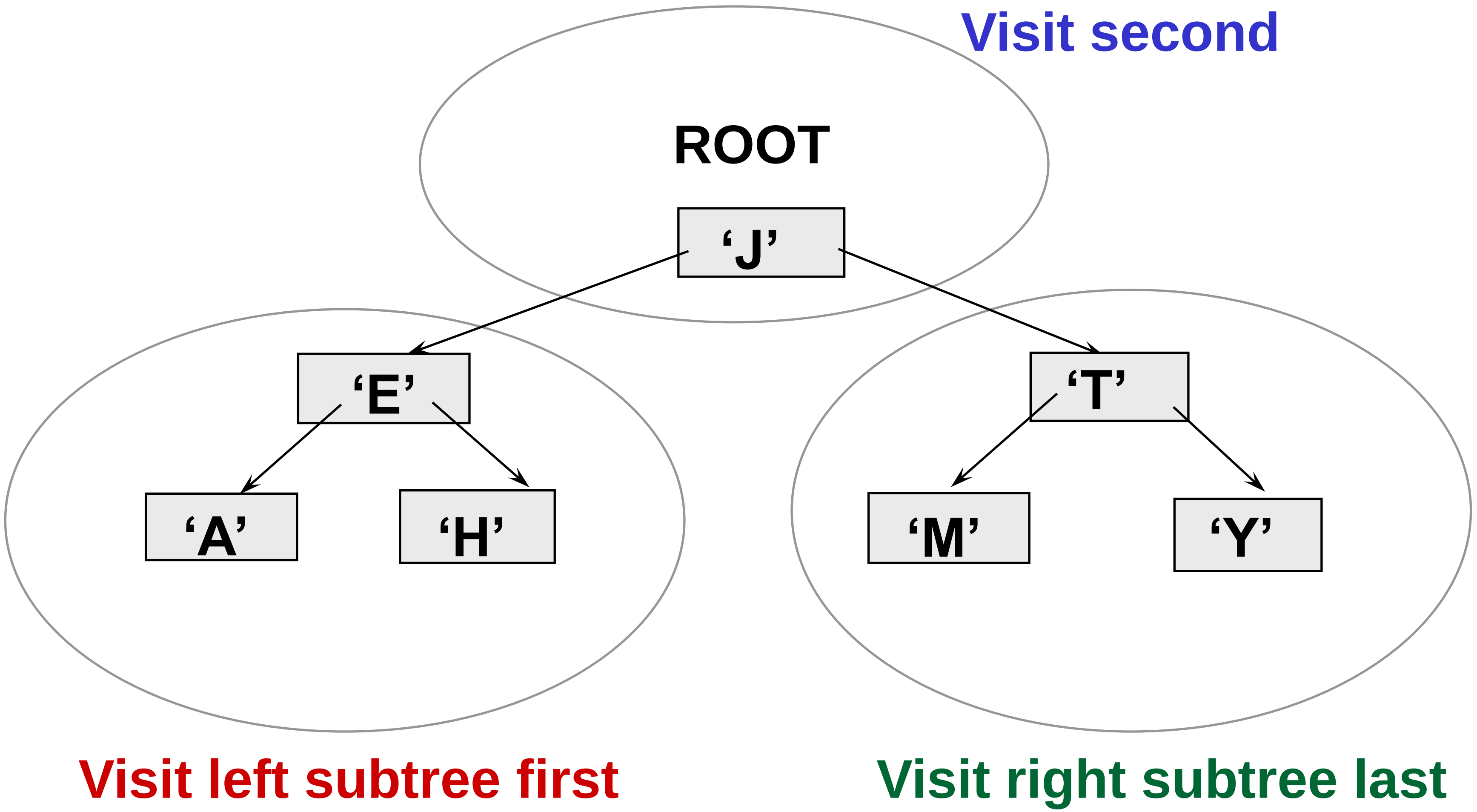
INORDER Traversal Algorithm

Let T be an ordered binary tree with root r.

If T has only r, then r is the inorder traversal.

Otherwise, suppose T_1, T_2 are the left and right subtrees at r. The inorder traversal begins by traversing T_1 in inorder. Then visits r, then traverses T_2 in inorder.

Inorder Traversal: A E H J M T Y



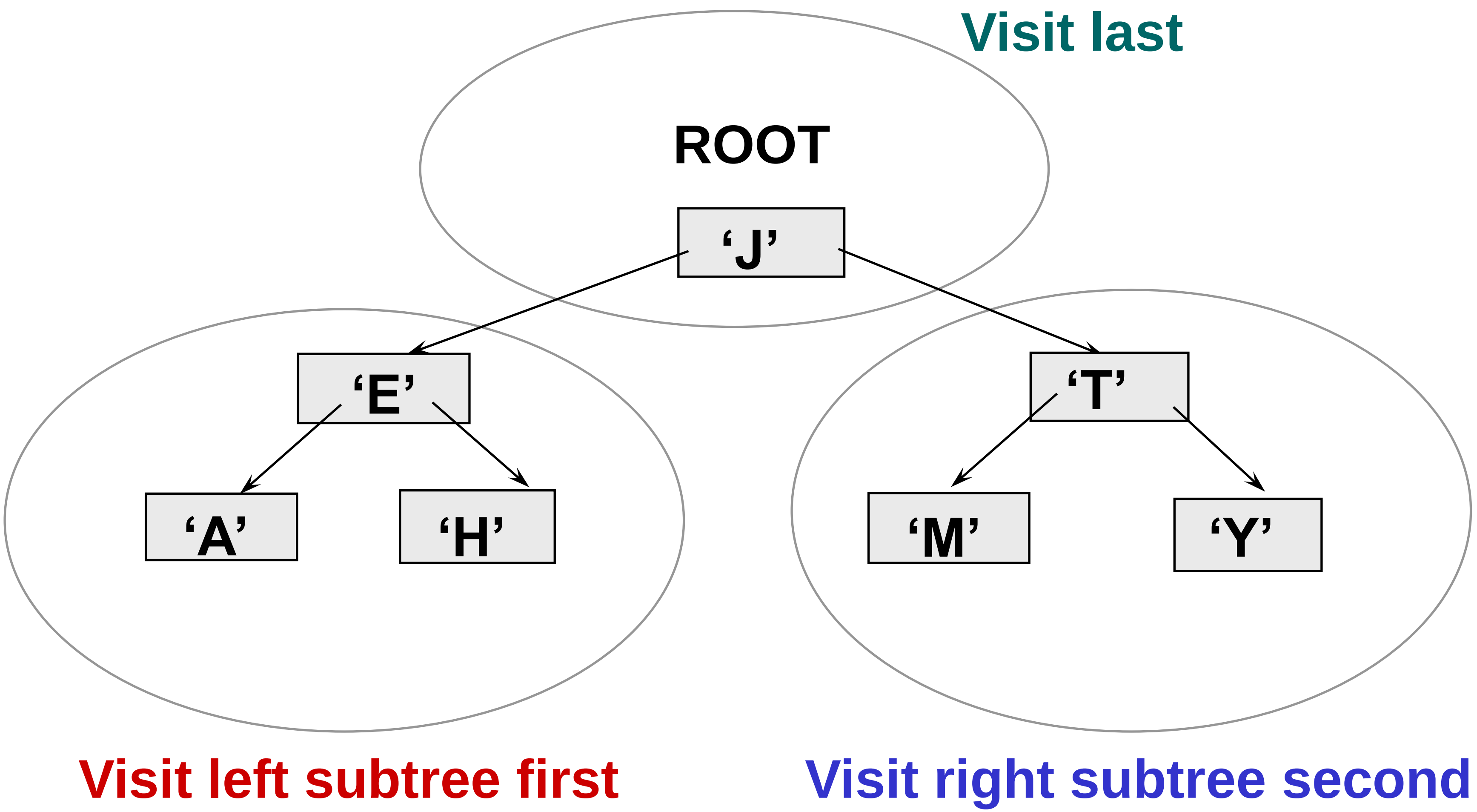
POSTORDER Traversal Algorithm

Let T be an ordered binary tree with root r.

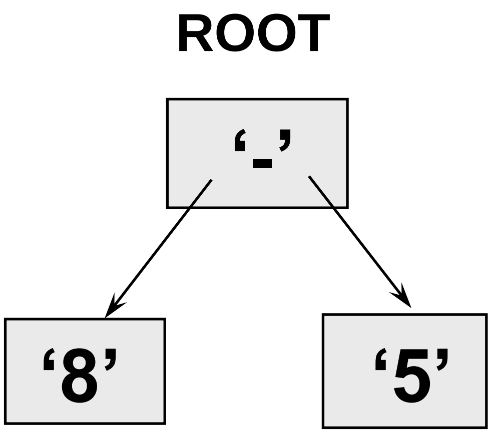
If T has only r, then r is the postorder traversal.

Otherwise, suppose T_1, T_2 are the left and right subtrees at r. The postorder traversal begins by traversing T_1 in postorder. Then traverses T_2 in postorder, then ends by visiting r.

Postorder Traversal: A H E M Y T J



A Binary Expression Tree



INORDER TRAVERSAL: `8 - 5 has value 3`

PREORDER TRAVERSAL: `- 8 5`

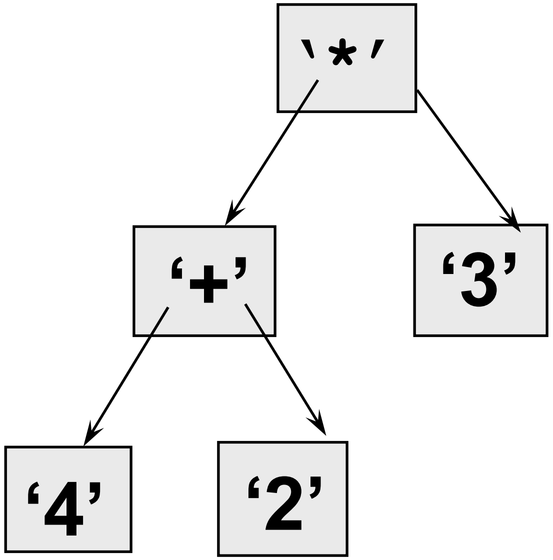
POSTORDER TRAVERSAL: `8 5 -`

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, and**
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.**

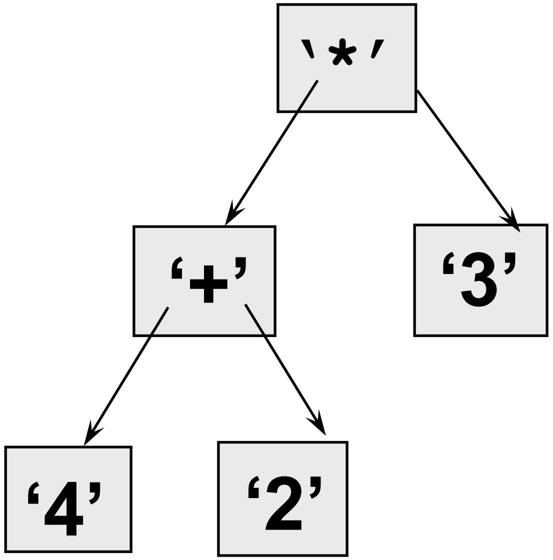
A Binary Expression Tree



What value does it have?

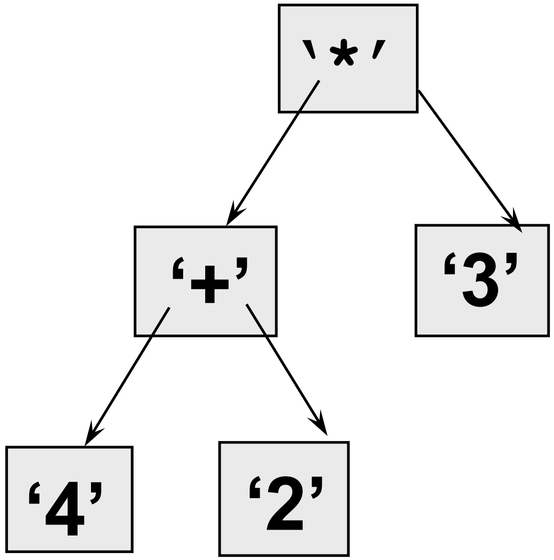
(4 + 2) * 3 = 18

A Binary Expression Tree



What infix, prefix, postfix expressions does it represent?

A Binary Expression Tree



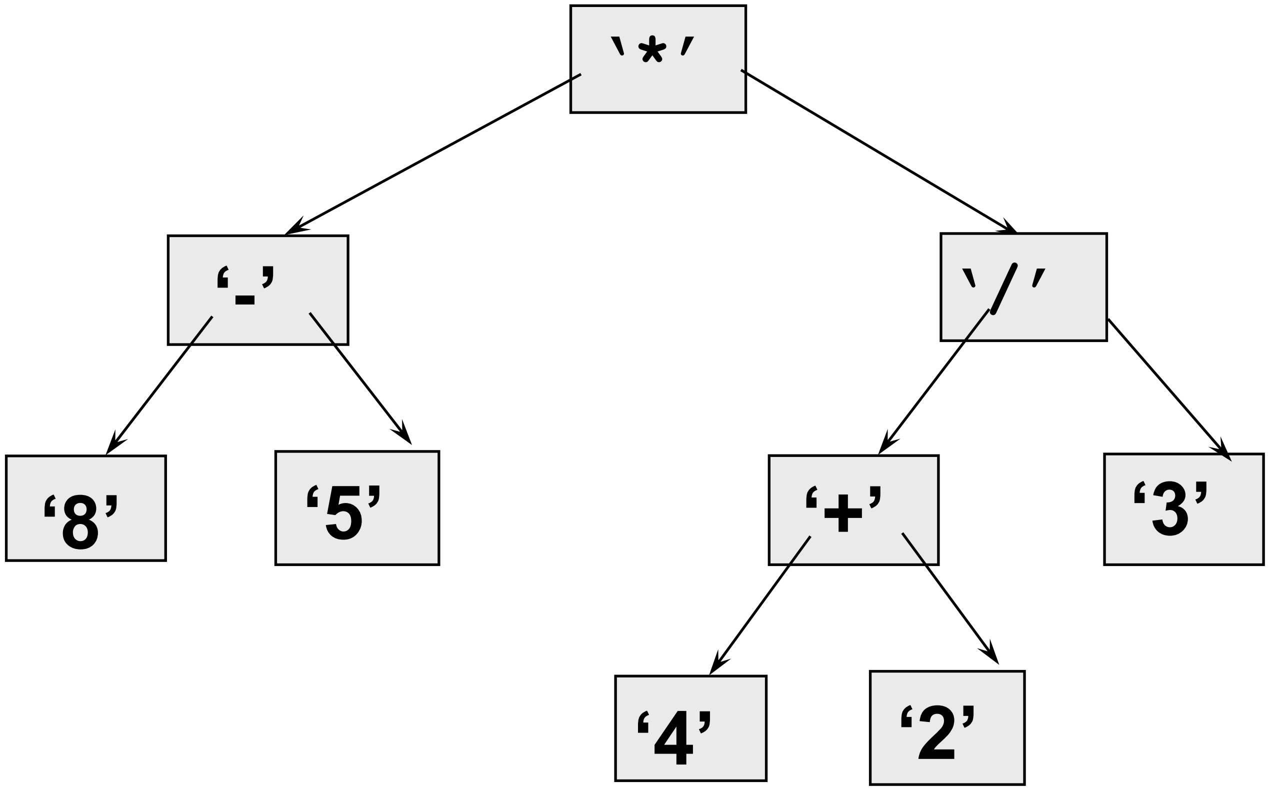
Infix:	((4 + 2) * 3)	
Prefix:	* + 4 2 3	*evaluate from right*
Postfix:	4 2 + 3 *	*evaluate from left*

Levels Indicate Precedence

When a binary expression tree is used to represent an expression, the levels of the nodes in the tree indicate their relative precedence of evaluation.

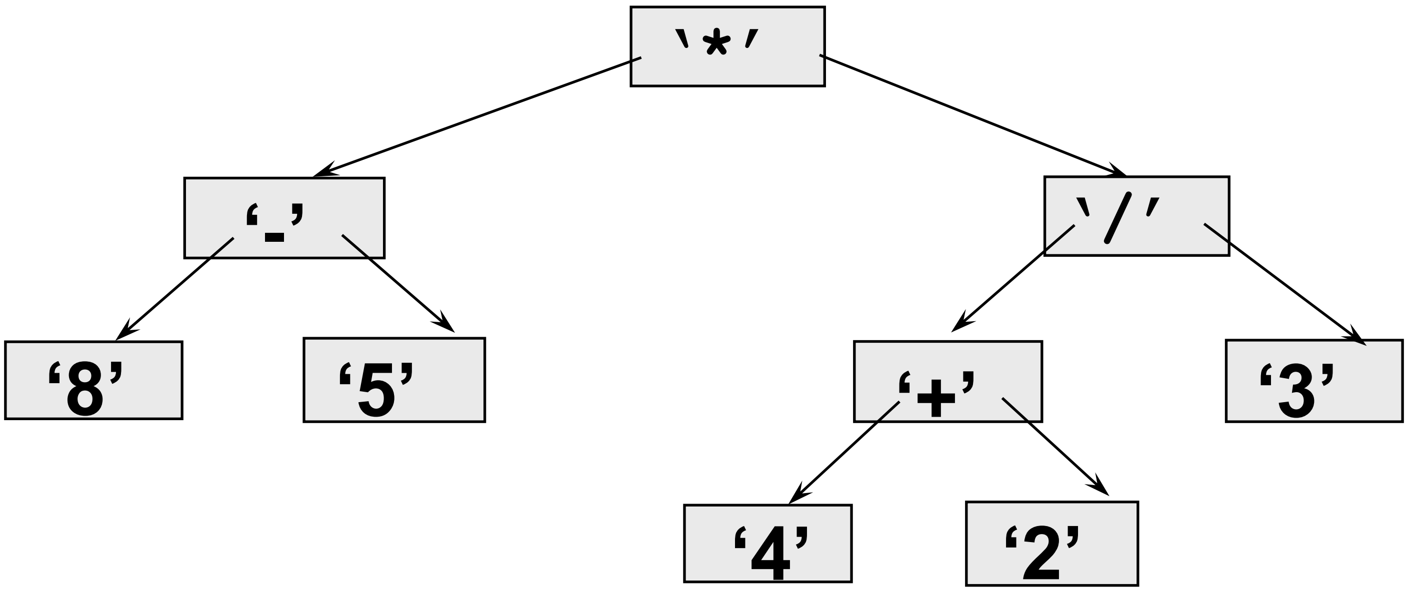
Operations at higher levels of the tree are evaluated later than those below them. The operation at the root is always the last operation performed.

Evaluate this binary expression tree



What infix, prefix, postfix expressions does it represent?

A binary expression tree

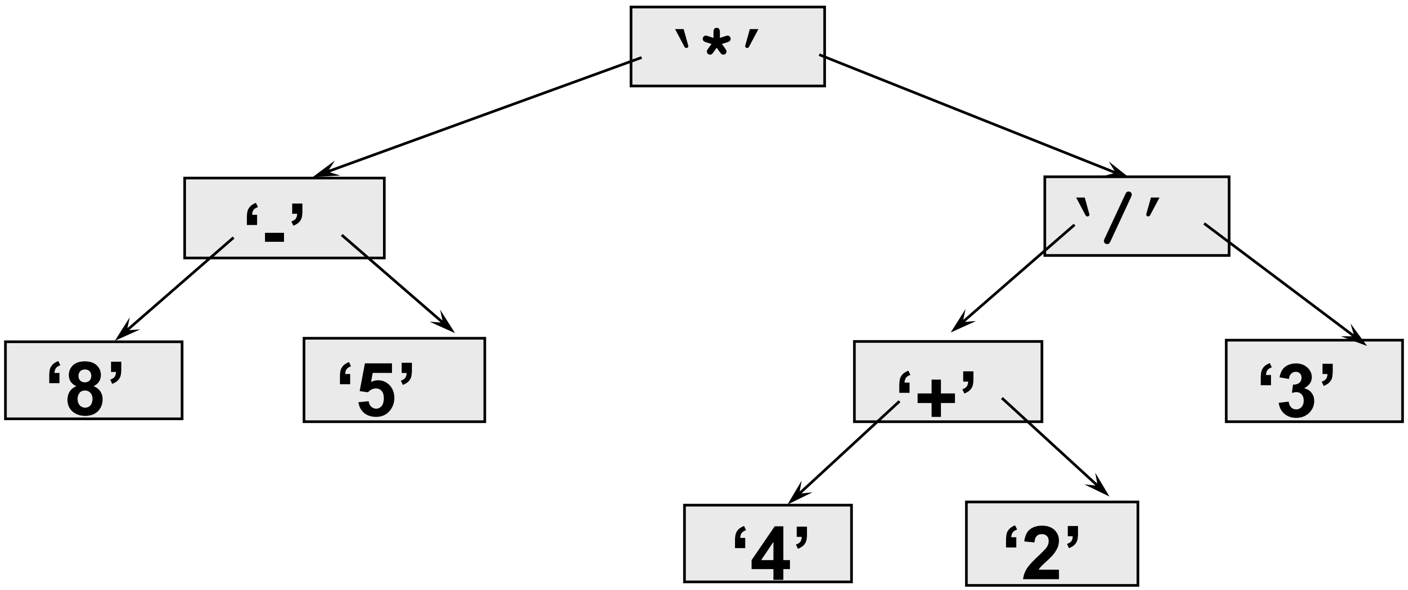


Infix: **((8 - 5) * ((4 + 2) / 3))**

Prefix: *** - 8 5 / + 4 2 3**

Postfix: **8 5 - 4 2 + 3 / *** ***has operators in order used***

A binary expression tree

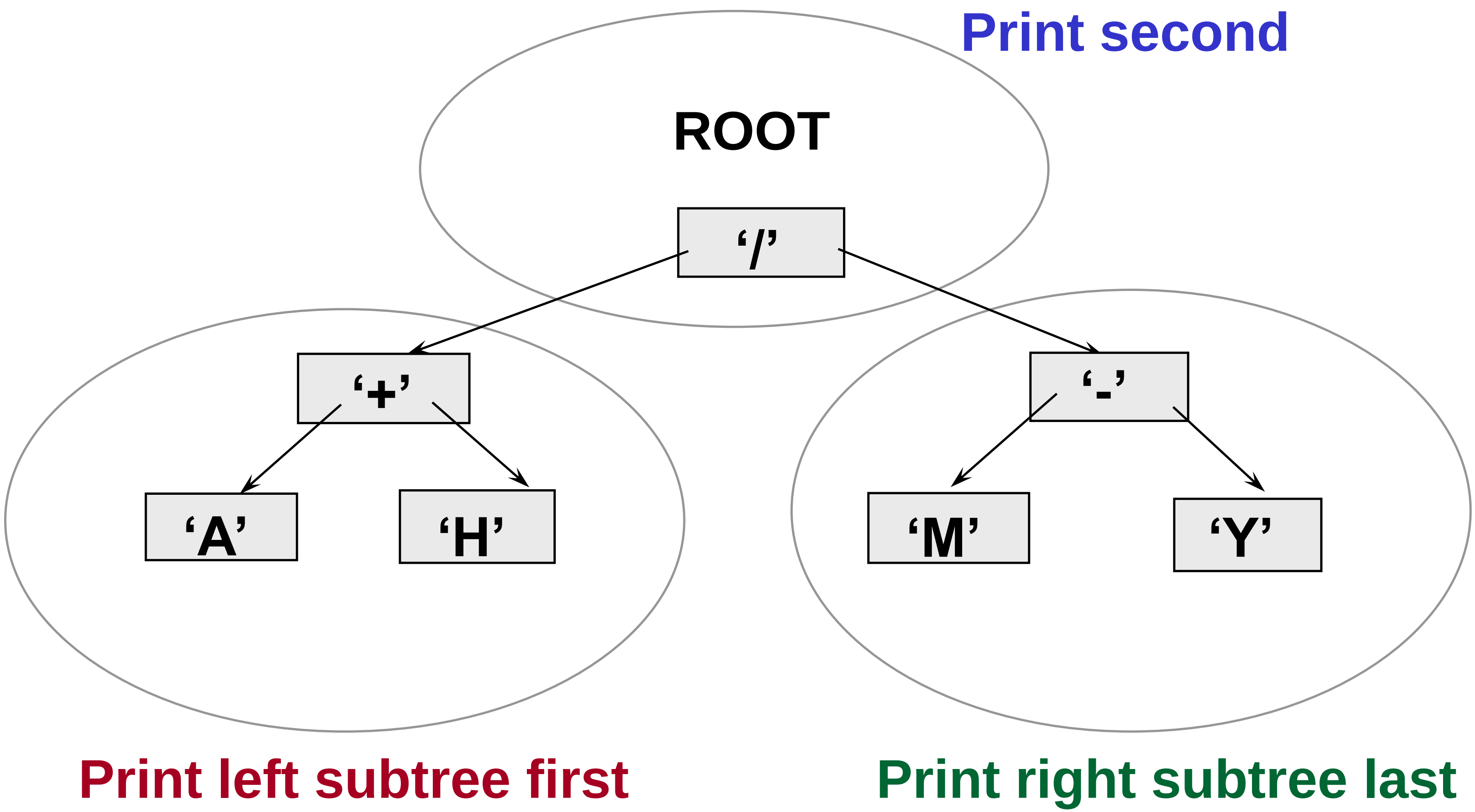


Infix: ((8 - 5) * ((4 + 2) / 3))

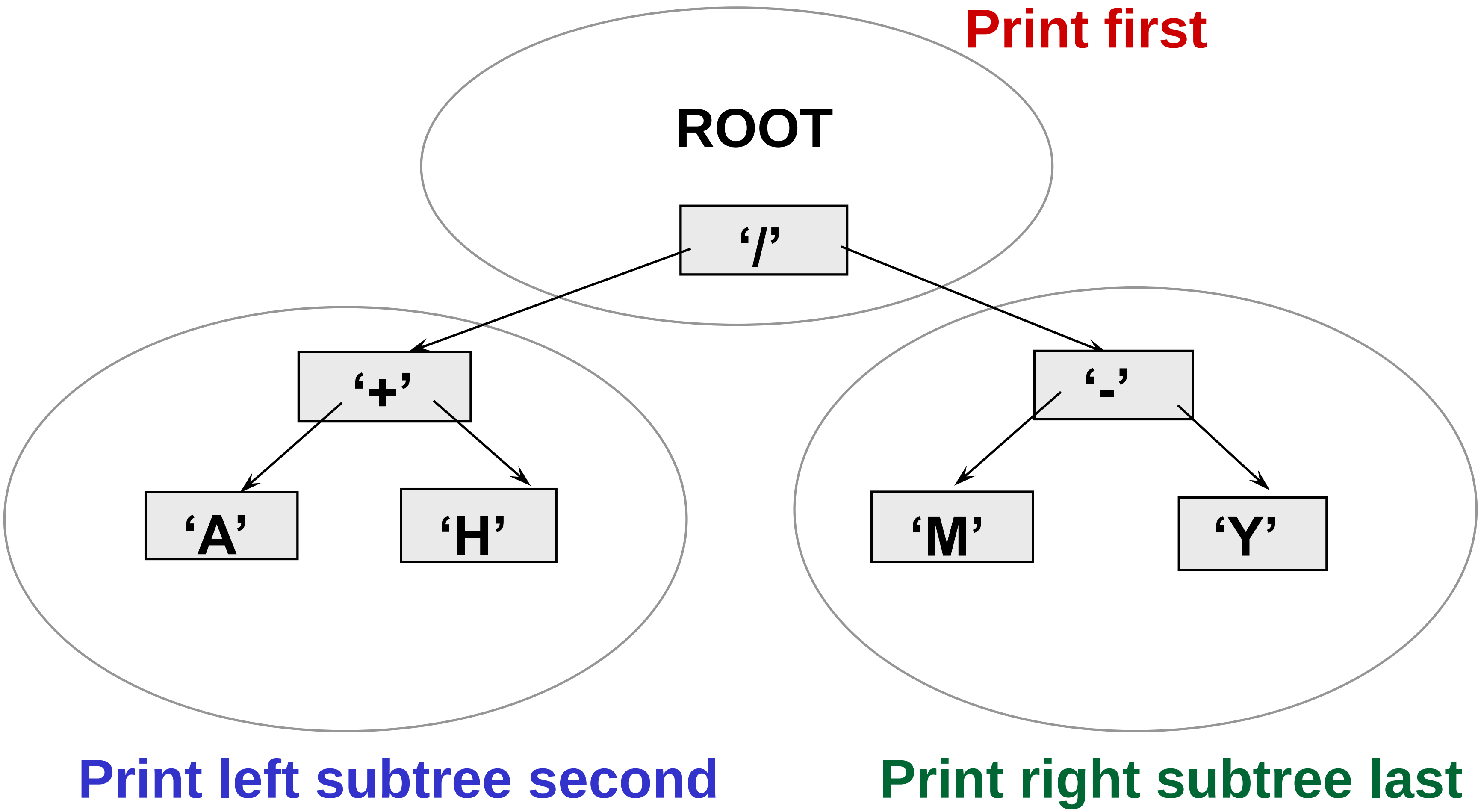
Prefix: * - 8 5 / + 4 2 3 *evaluate from right*

Postfix: 8 5 - 4 2 + 3 / * *evaluate from left*

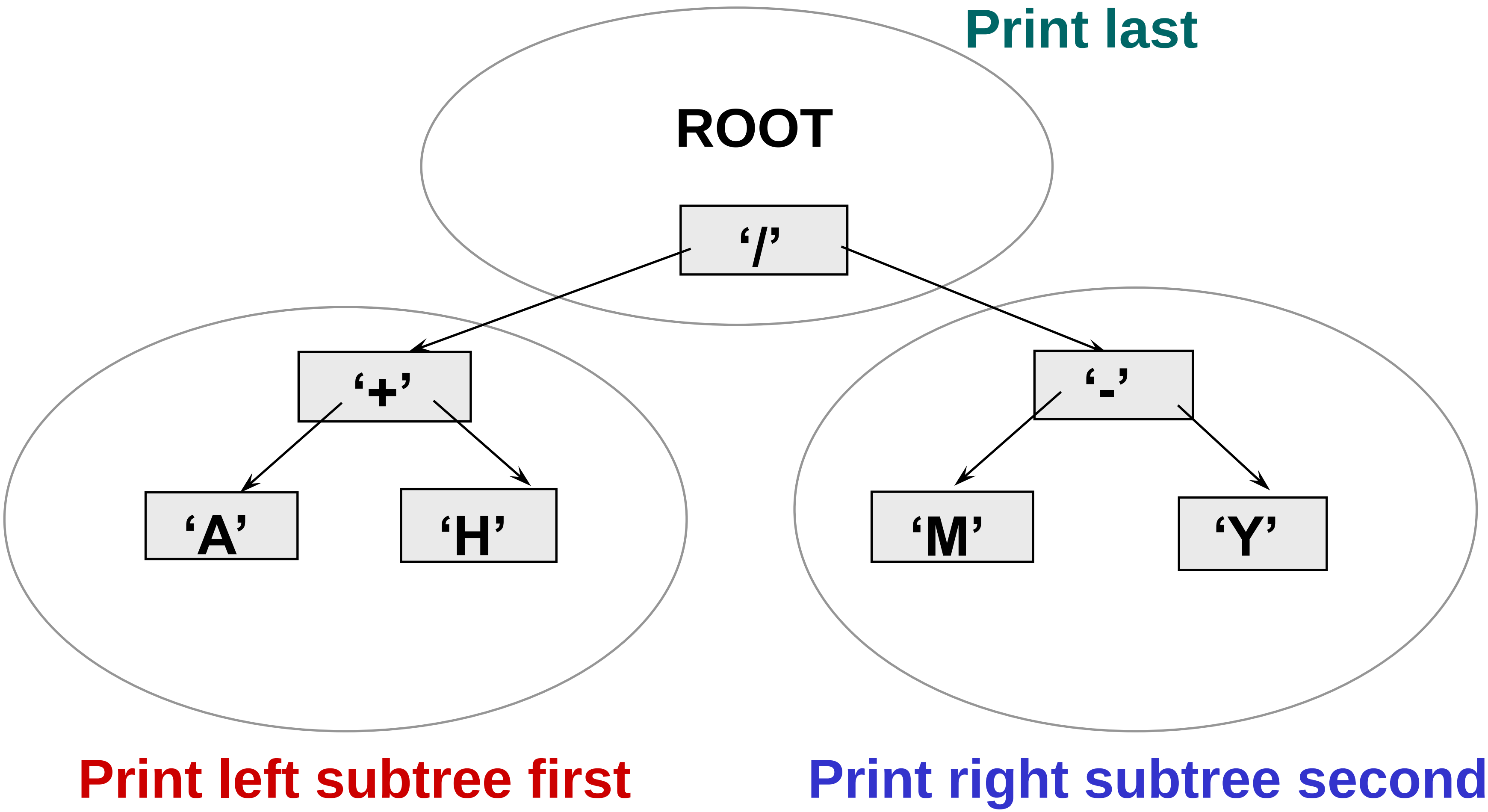
Inorder Traversal: (A + H) / (M - Y)



Preorder Traversal: / + A H - M Y



Postorder Traversal: A H + M Y - /



The End