

# COMP 250

## INTRODUCTION TO COMPUTER SCIENCE

Assignment Project Exam Help

<https://powcoder.com>

Week 11-1: Rooted Trees

Giulia Alberini, Fall 2020

Slides adapted from Michael Langer's

# WHAT ARE WE GOING TO DO IN THIS VIDEO?



- **Rooted Trees** Assignment Project Exam Help

- Terminology

- Implementation

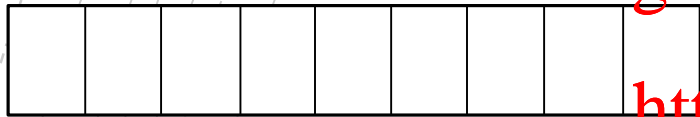
<https://powcoder.com>

Add WeChat powcoder

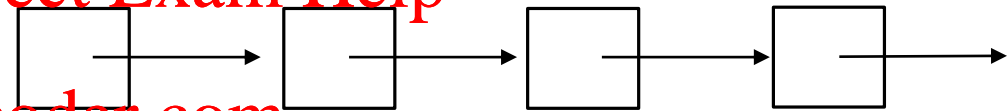
# DATA STRUCTURES

## ■ Linear

array



Linked list

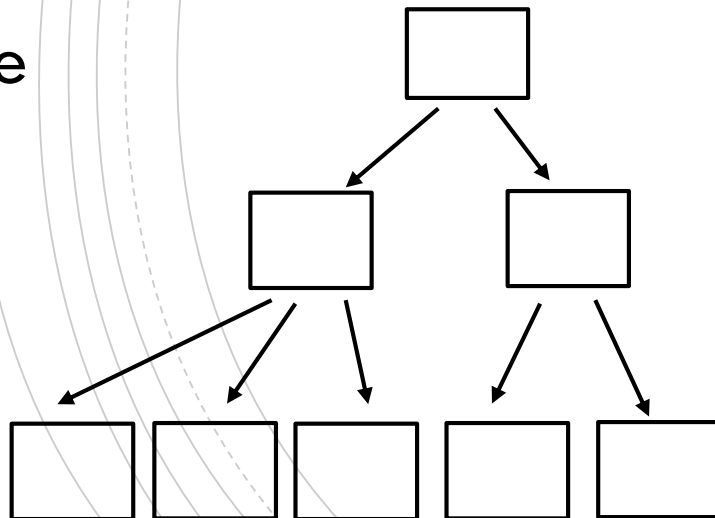


Assignment Project Exam Help

<https://powcoder.com>

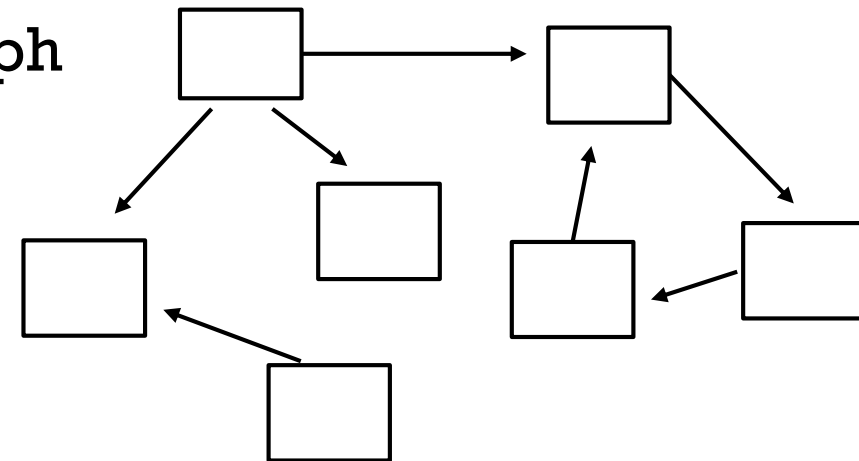
## ■ Non-linear

tree



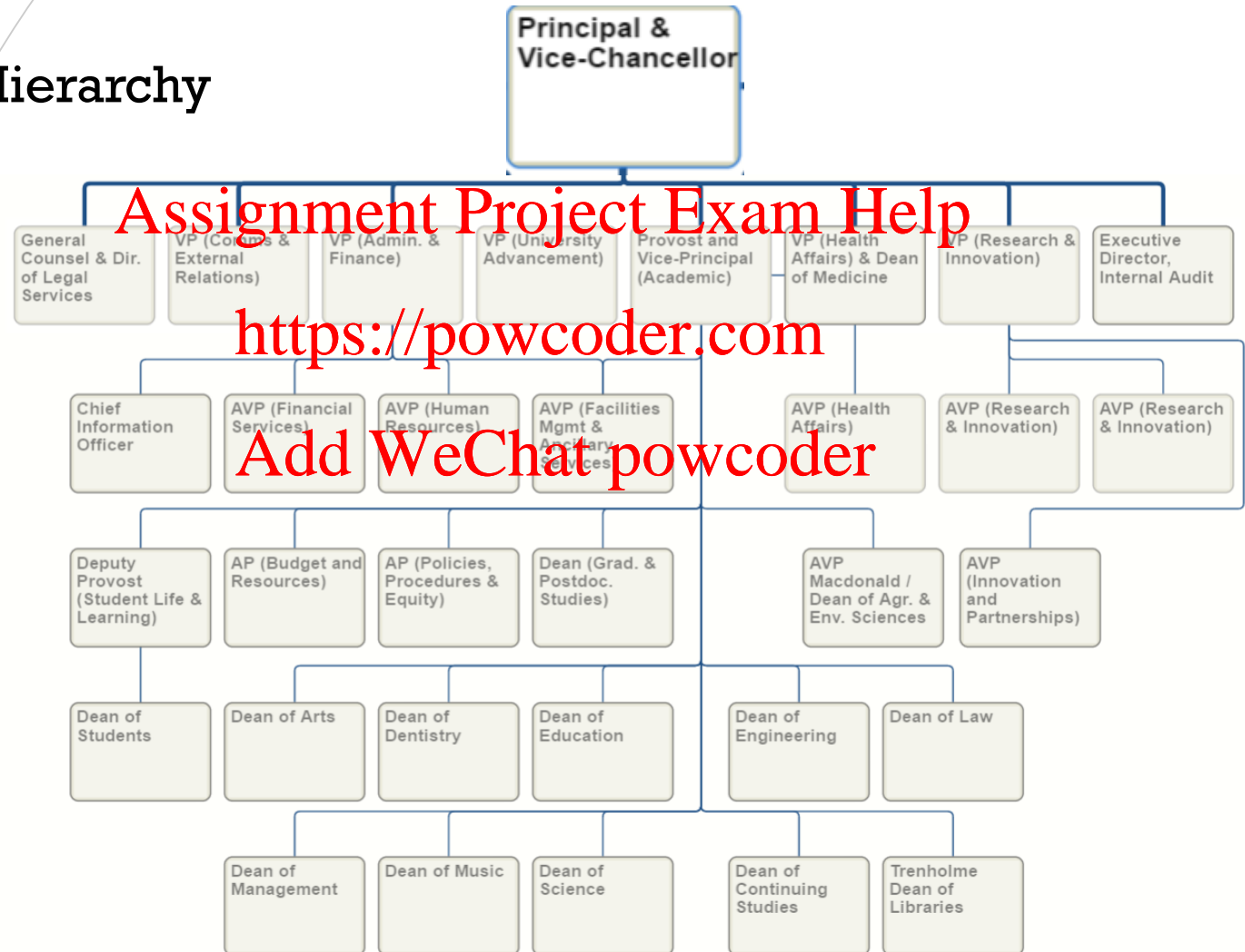
Add WeChat powcoder

graph

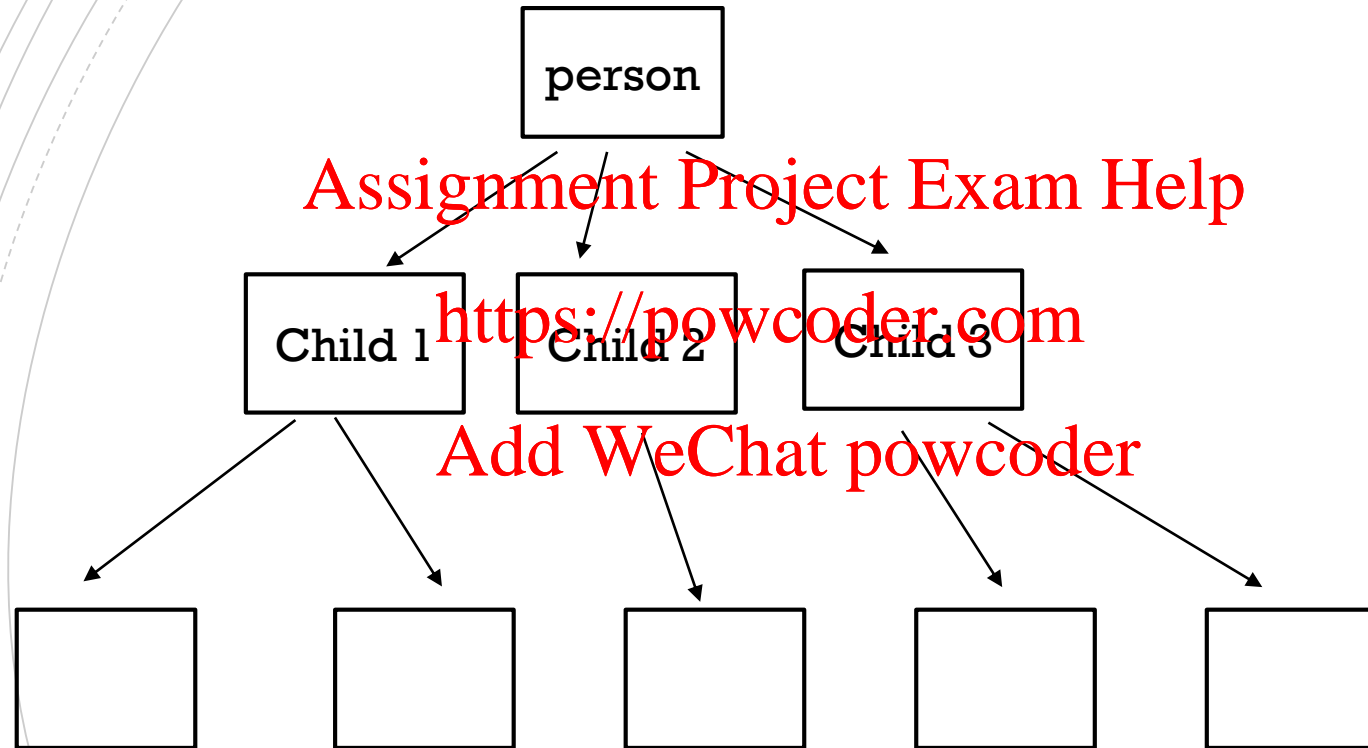


# TREE - EXAMPLE

- Organizational Hierarchy (McGill)

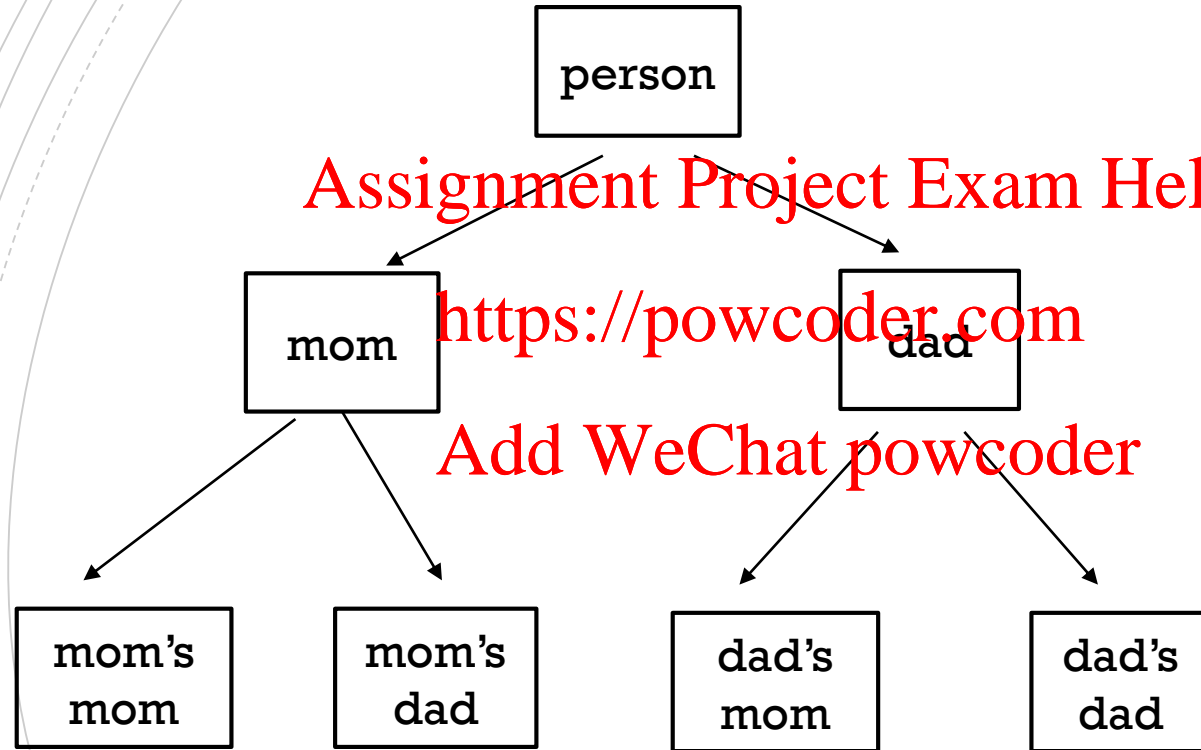


## EXAMPLE 2: FAMILY TREE (DESCENDANTS)



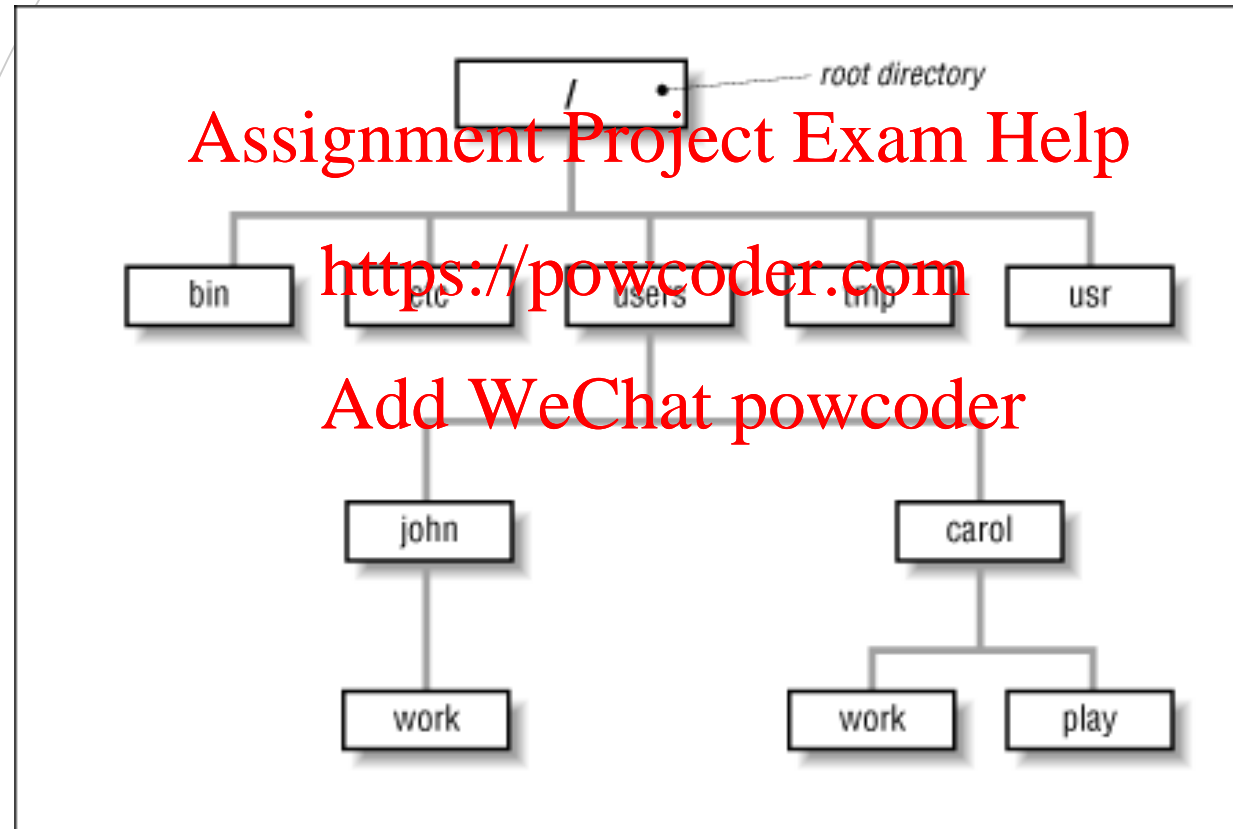
Here we ignore spouses (partner).

## EXAMPLE 3: FAMILY TREE (ANCESTORS)

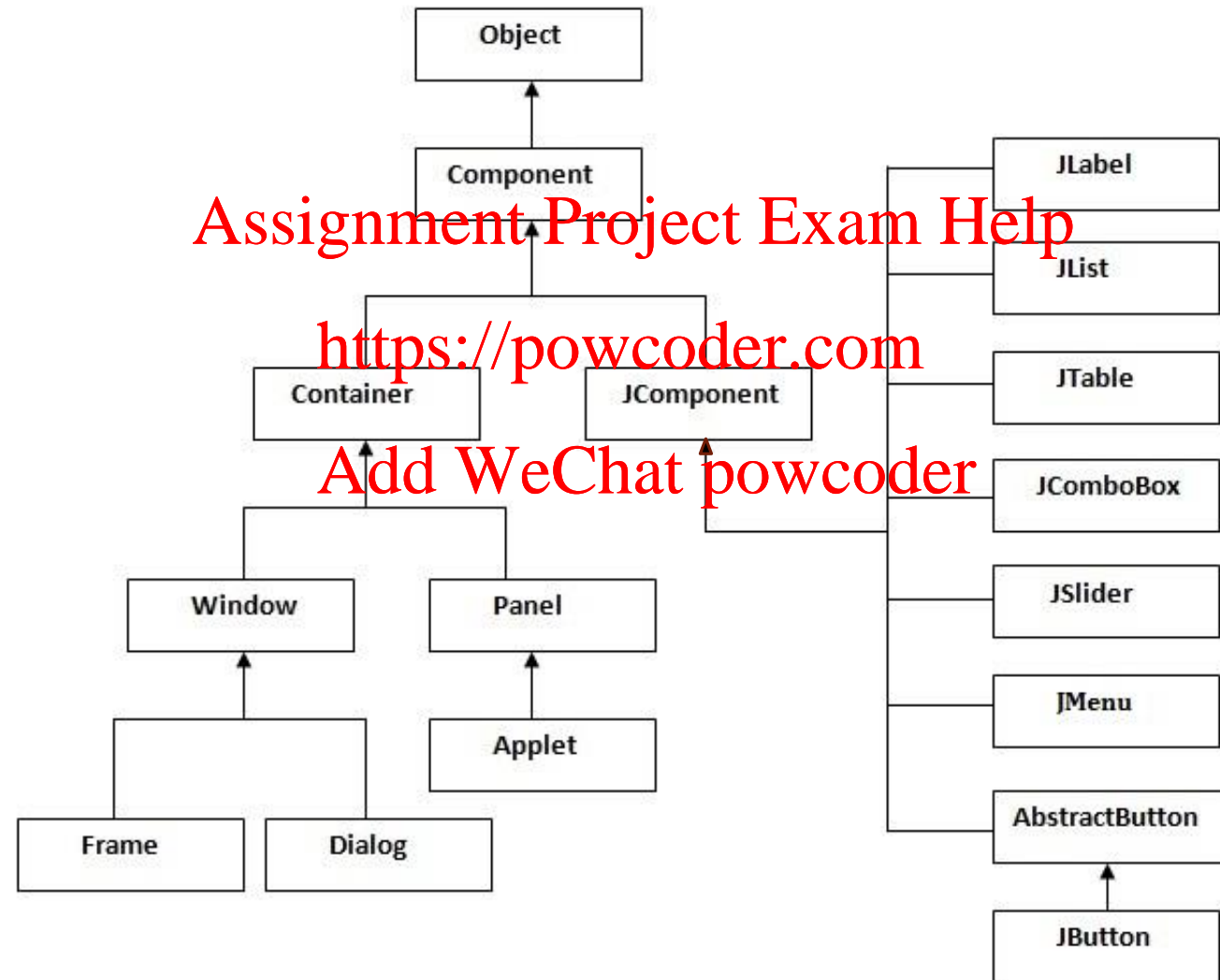


This is an example of a binary tree.

## EXAMPLE 4: UNIX FILE SYSTEM



## EXAMPLE 5: JAVA CLASSES E.G. GUI



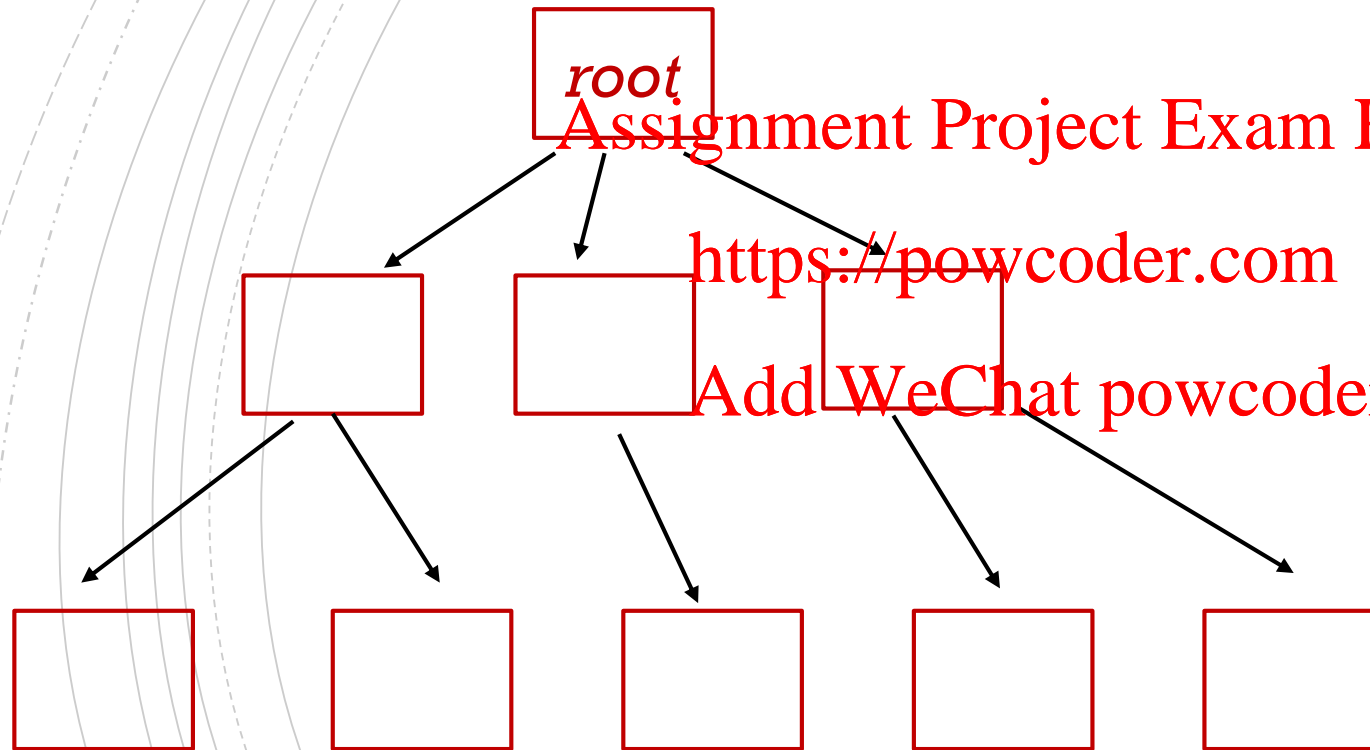


Assignment Project Exam Help

DEFINITIONS  
<https://powcoder.com>

Add WeChat powcoder

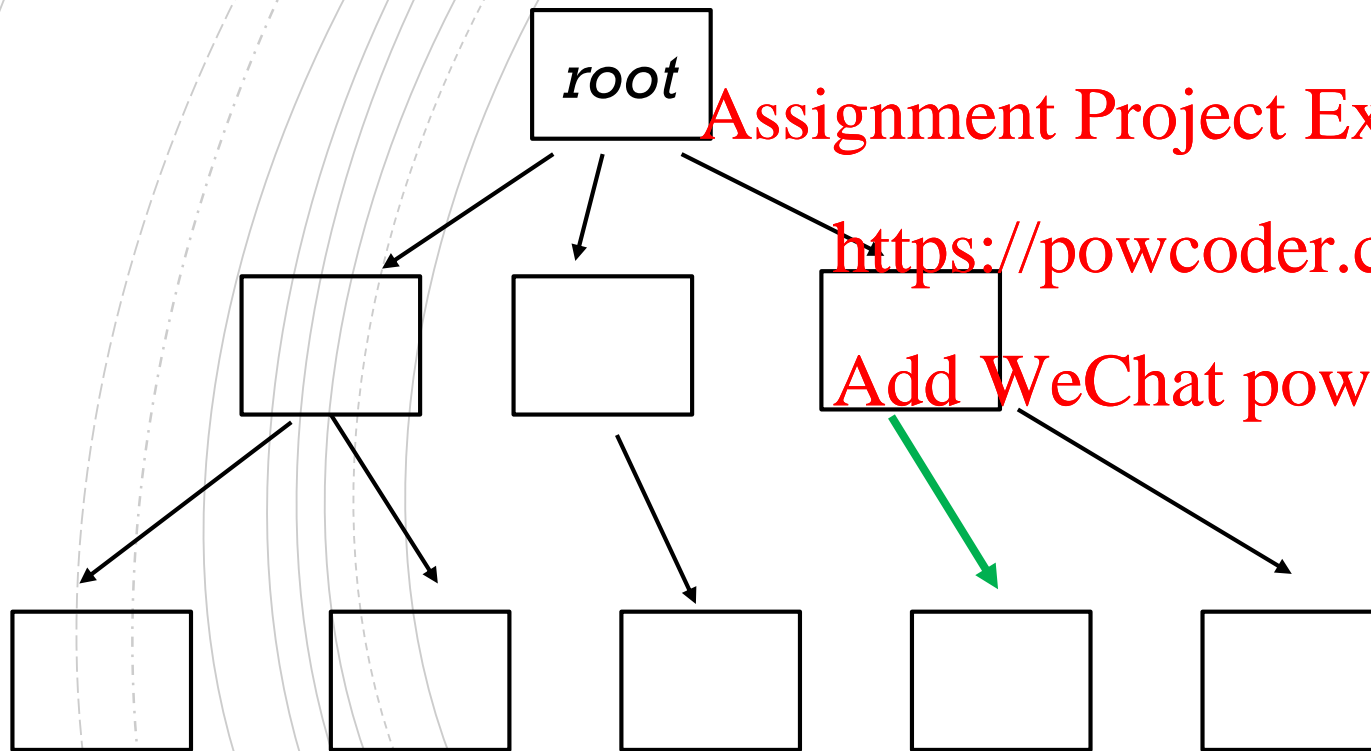
## (ROOTED) TREE TERMINOLOGY



A tree is a collection of **nodes** (*vertexes*)

The **root** is the top node in a tree

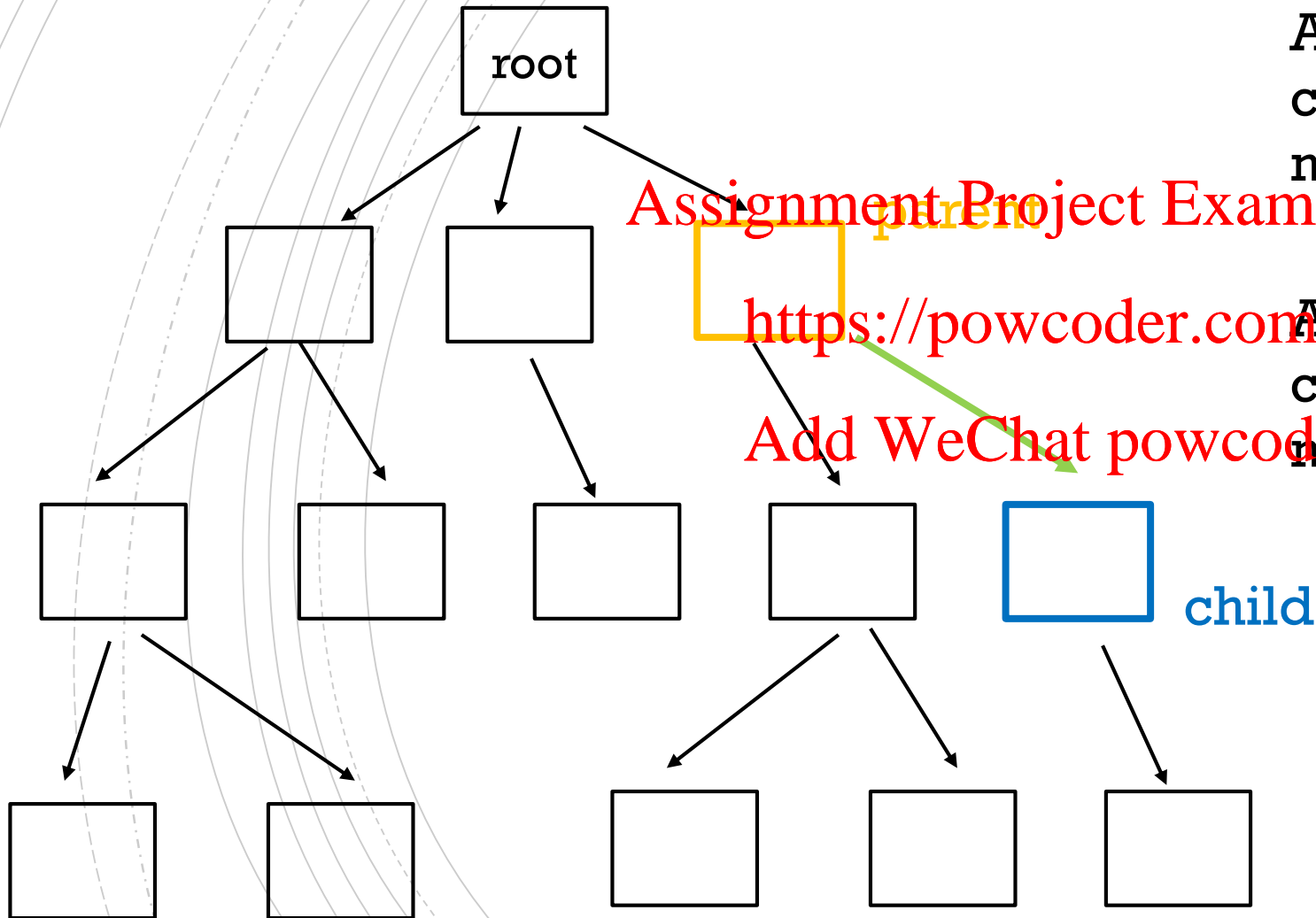
## (ROOTED) TREE TERMINOLOGY



A **directed edge** is ordered pair of nodes  $(v_i, v_j)$  (from, to).

Trees can be *undirected* or *directed*. If directed, the edges are either all pointing away from the root or all pointing towards the root.

## (ROOTED) TREE TERMINOLOGY



A *child* is a node directly connected to another node when moving away from the root.

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

A *parent* is a node directly connected to another node when moving towards the root.

Every node except the root is a *child*, and has exactly one *parent*.

## EDGE DIRECTION

For some trees,

- edges are directed from parent to child
- edges are directed from child to parent
- edges are directed both from parent to child and child to parent.
- edge direction is ignored e.g. common with non-rooted trees (see next slide)

*Most of definitions today will assume edges are from parent to child.*

## ASIDE: NON-ROOTED TREES

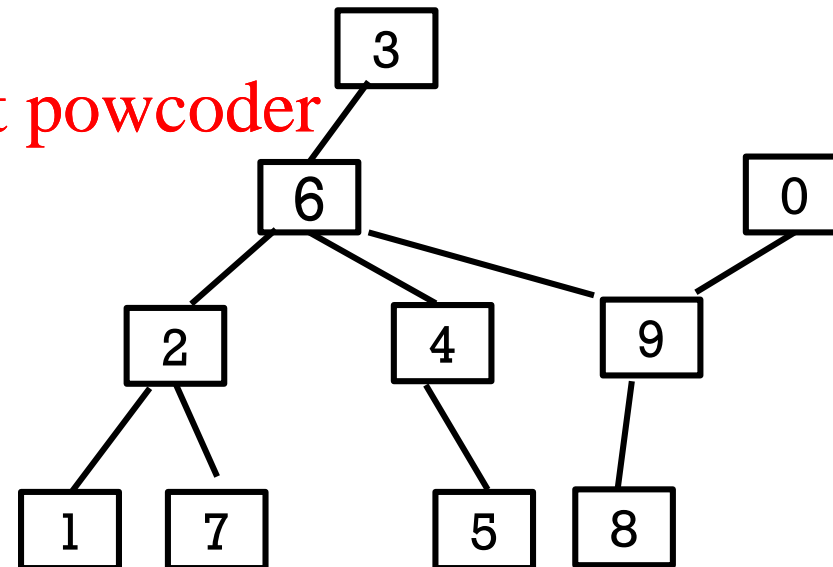
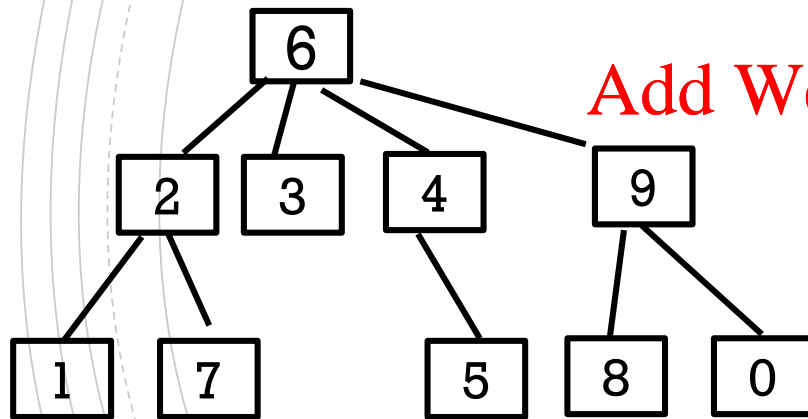
You will see non-rooted trees most commonly when edges are undirected, and there is no natural way to define the 'root'.

You will see examples in COMP 251.

e.g. the tree on the left is only rooted because I drew it that way. It is actually the same (non-rooted) tree as the one on the right.

<https://powcoder.com>

Add WeChat powcoder



## NUMBER OF EDGES

- Q: If a (rooted) tree has  $n$  nodes, then how many edges does it have?

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

## NUMBER OF EDGES

- Q: If a (rooted) tree has  $n$  nodes, then how many edges does it have?

Assignment Project Exam Help

- A:  $n - 1$

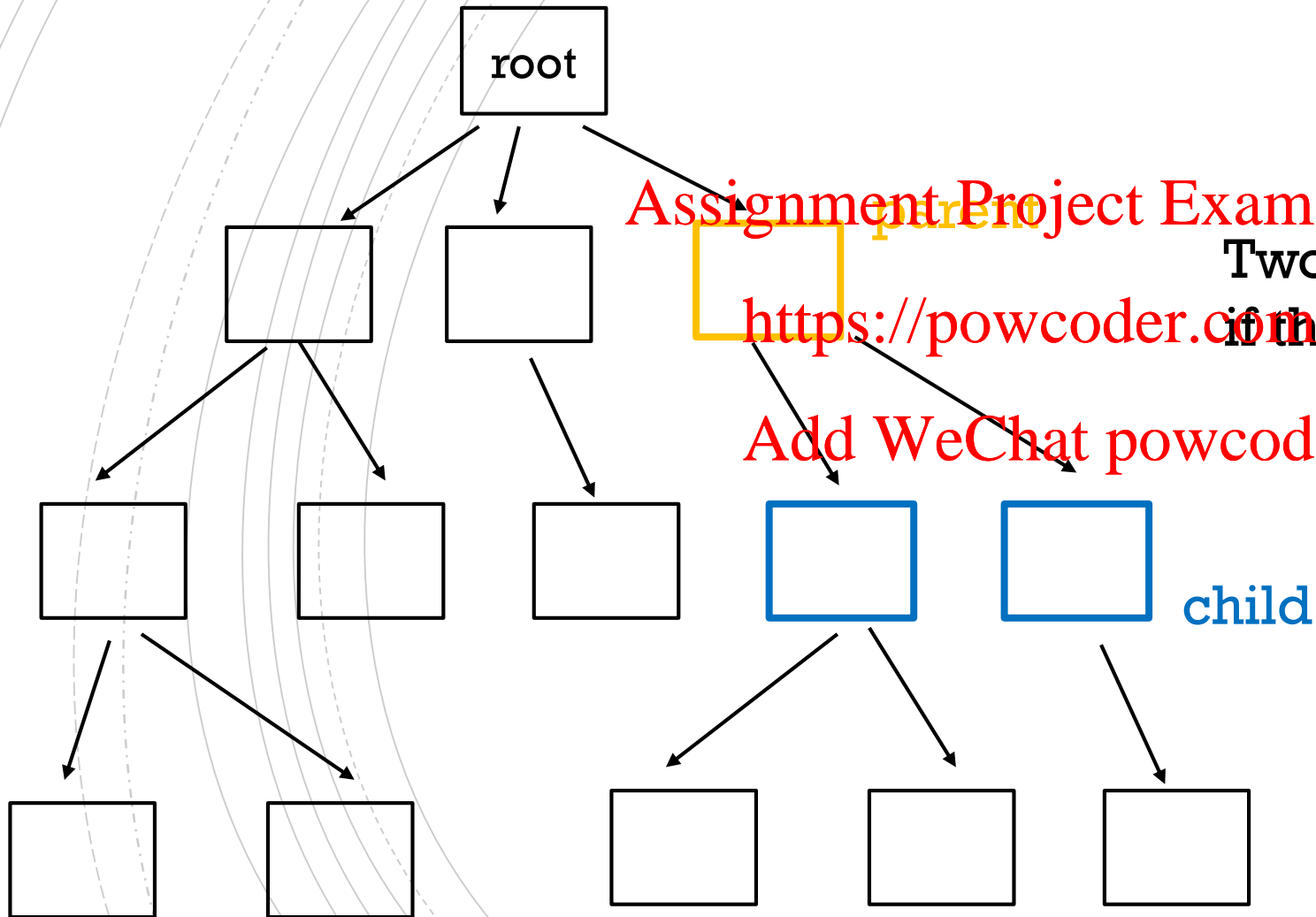
<https://powcoder.com>

Add WeChat powcoder

Since every edge is of the form (parent, child), and each node except the root is a child and each child has exactly one parent.



## (ROOTED) TREE TERMINOLOGY



Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Two nodes are said to be **siblings** if they have the same **parent**.

child

## RECURSIVE DEFINITION OF ROOTED TREE

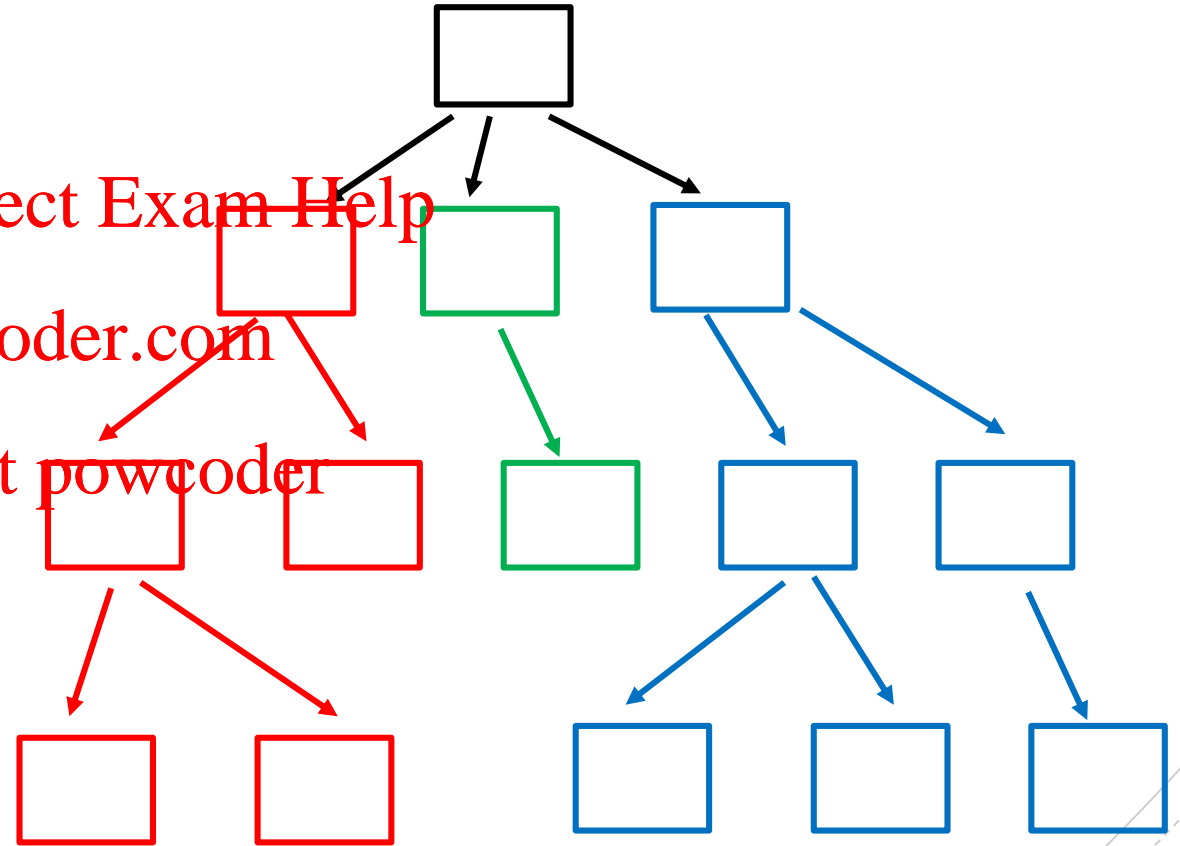
A tree  $T$  is a finite (& possibly empty) set of  $n$  nodes such that:

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

?

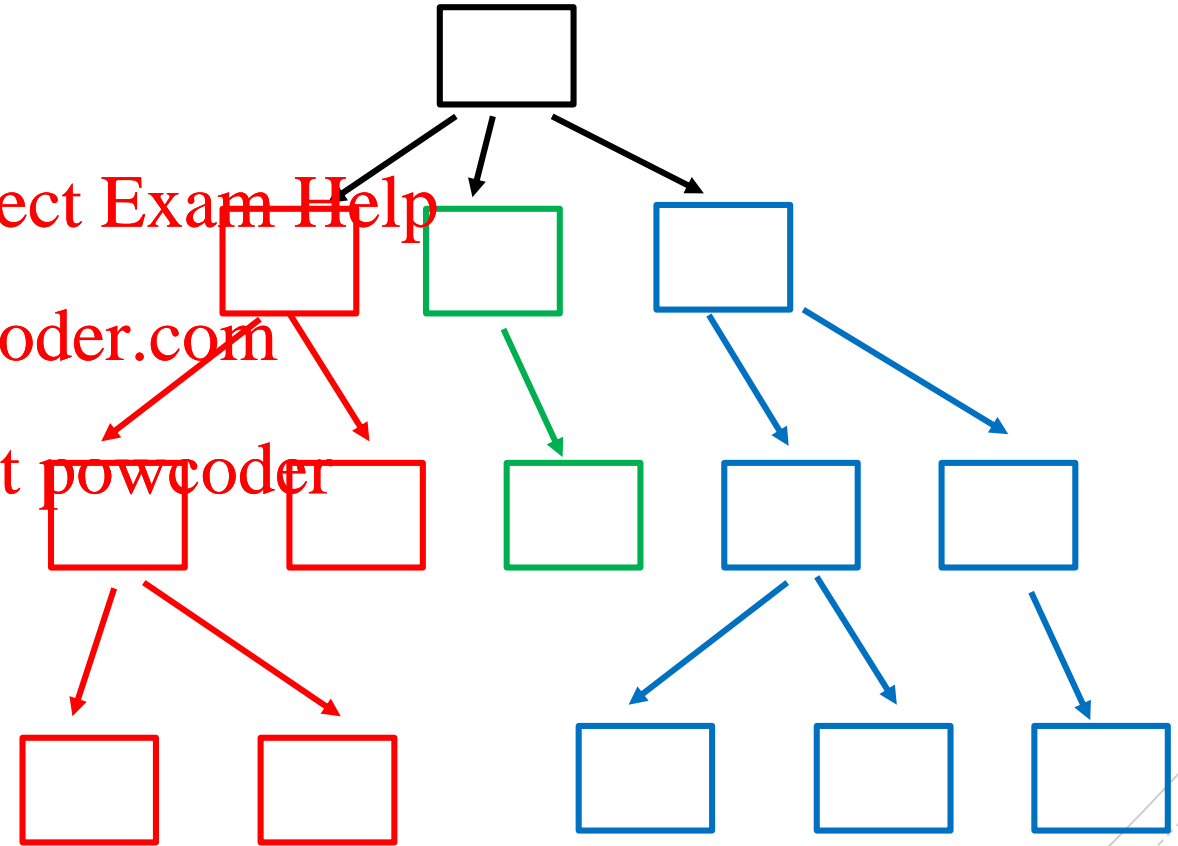


## RECURSIVE DEFINITION OF ROOTED TREE

A tree  $T$  is a finite (& possibly empty) set of  $n$  nodes such that:

- if  $n > 0$  then one of the nodes is the root

?



Assignment Project Exam Help

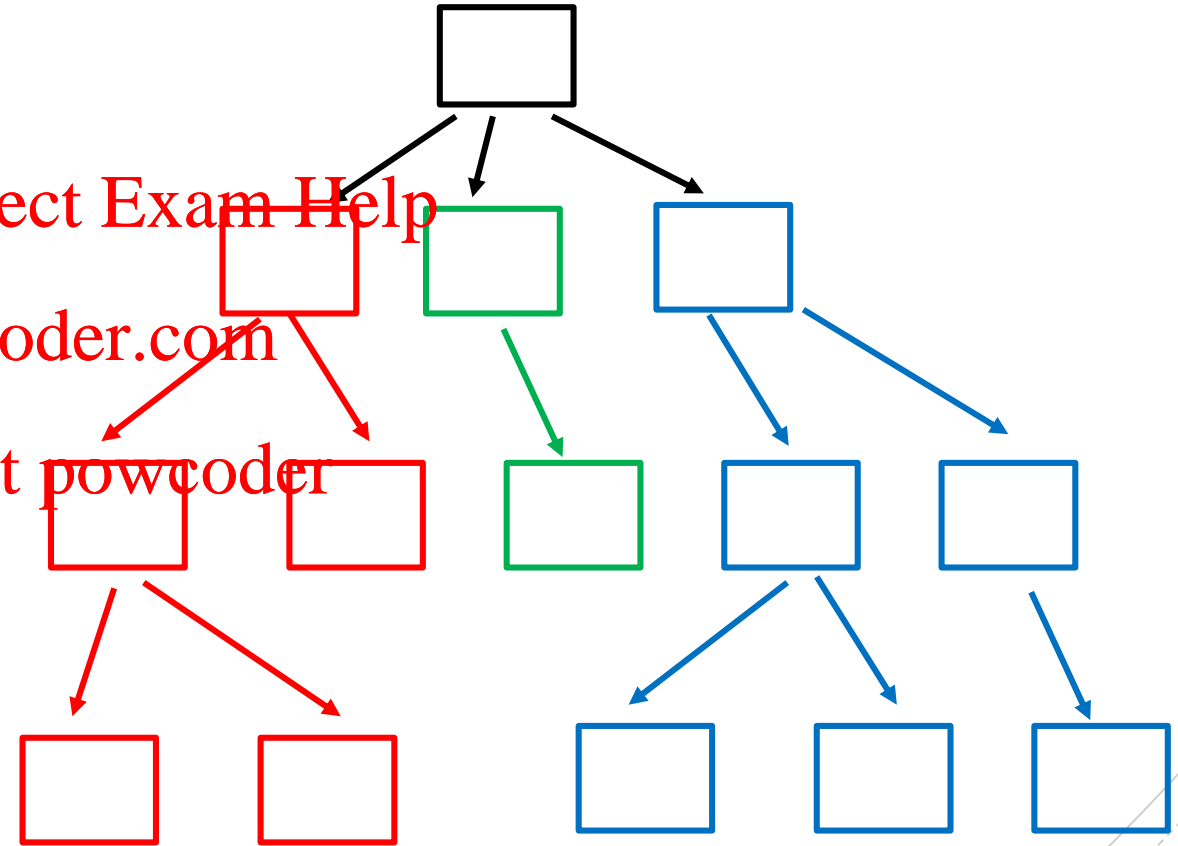
<https://powcoder.com>

Add WeChat powcoder

## RECURSIVE DEFINITION OF ROOTED TREE

A tree  $T$  is a finite (& possibly empty) set of  $n$  nodes such that:

- if  $n > 0$  then one of the nodes is the root
- if  $n > 1$  then the  $n - 1$  non-root nodes are partitioned into (non-empty) subsets  $T_1, T_2, \dots, T_k$ , each of which is a tree (called a “subtree”), and the roots of the subtrees are the children of root  $r$ .



Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

## ANOTHER DEFINITION

A recursive definition for tree can also be given using lists as follows:

**Assignment Project Exam Help**  
**<https://powcoder.com>**  
**Add WeChat powcoder**

```
tree      =    root | ( root listOfSubTrees )  
listOfSubTrees = tree | tree listOfSubTrees
```

**Note that** `listOfSubTrees` **cannot be empty.**

## TRY IT!

A recursive definition for tree can also be given using lists as follows:

**Assignment Project Exam Help**  
**<https://powcoder.com>**  
**Add WeChat powcoder**

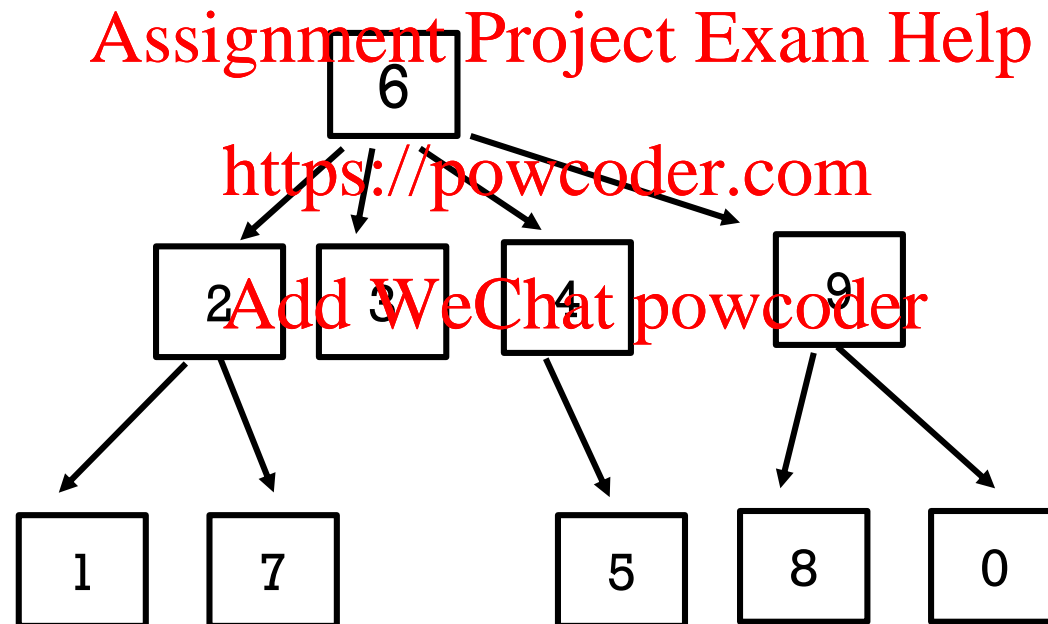
```
tree      =      root      |      ( root listOfSubTrees )  
listOfSubTrees = tree | tree listOfSubTrees
```

- Draw the tree that corresponds to the following list, where the root elements are single digits.

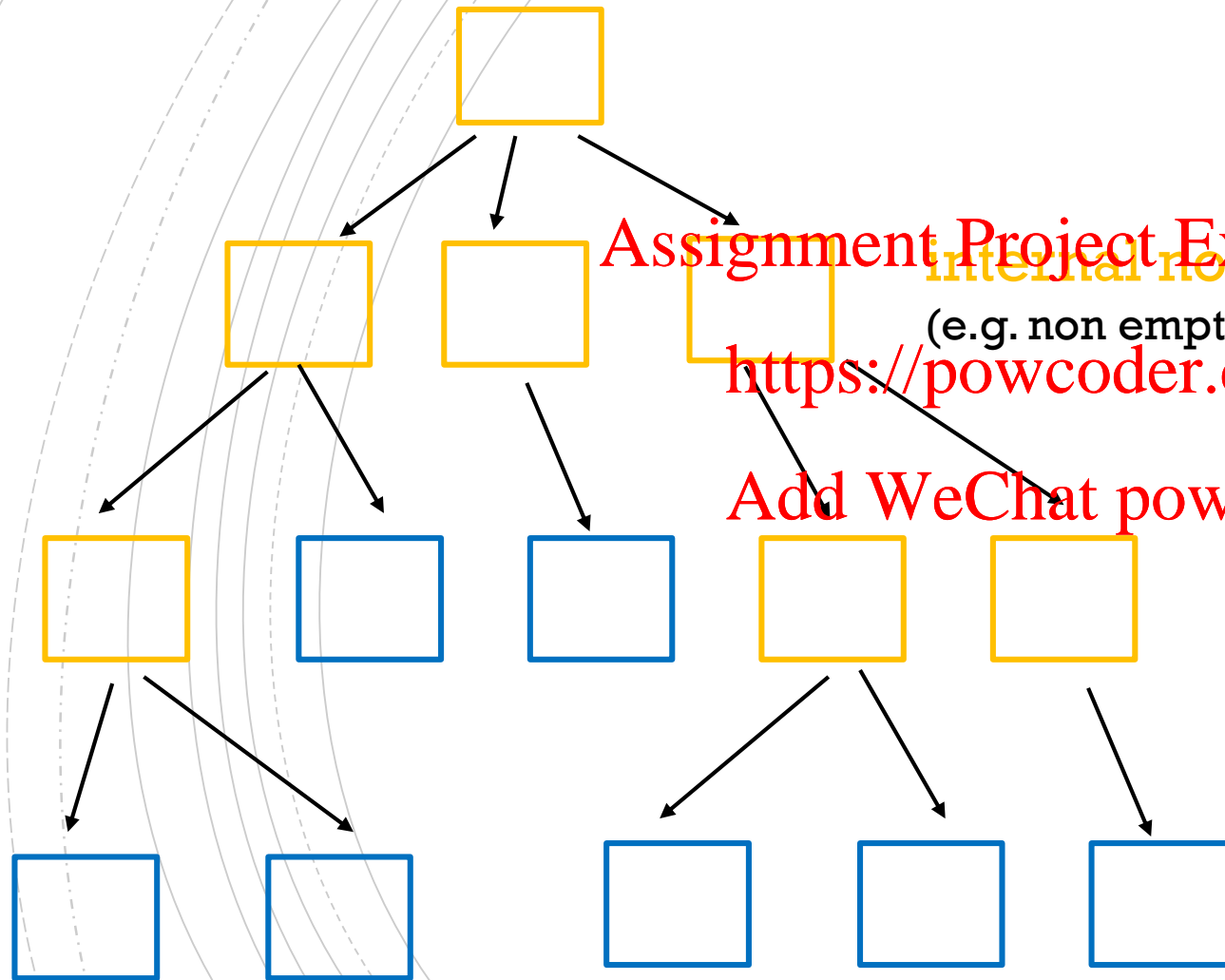
( 6 ( 2 1 7 ) 3 ( 4 5 ) ( 9 8 0 ) )

## SOLUTION

$(6 \ (2 \ 1 \ 7) \ 3 \ (4 \ 5) \ (9 \ 8 \ 0))$  represents the following tree:



## (ROOTED) TREE TERMINOLOGY



An *internal node* is a node with at least one child.

Assignment Project Exam Help

Internal nodes

(e.g. non empty file directories)

<https://powcoder.com>

Add WeChat powcoder

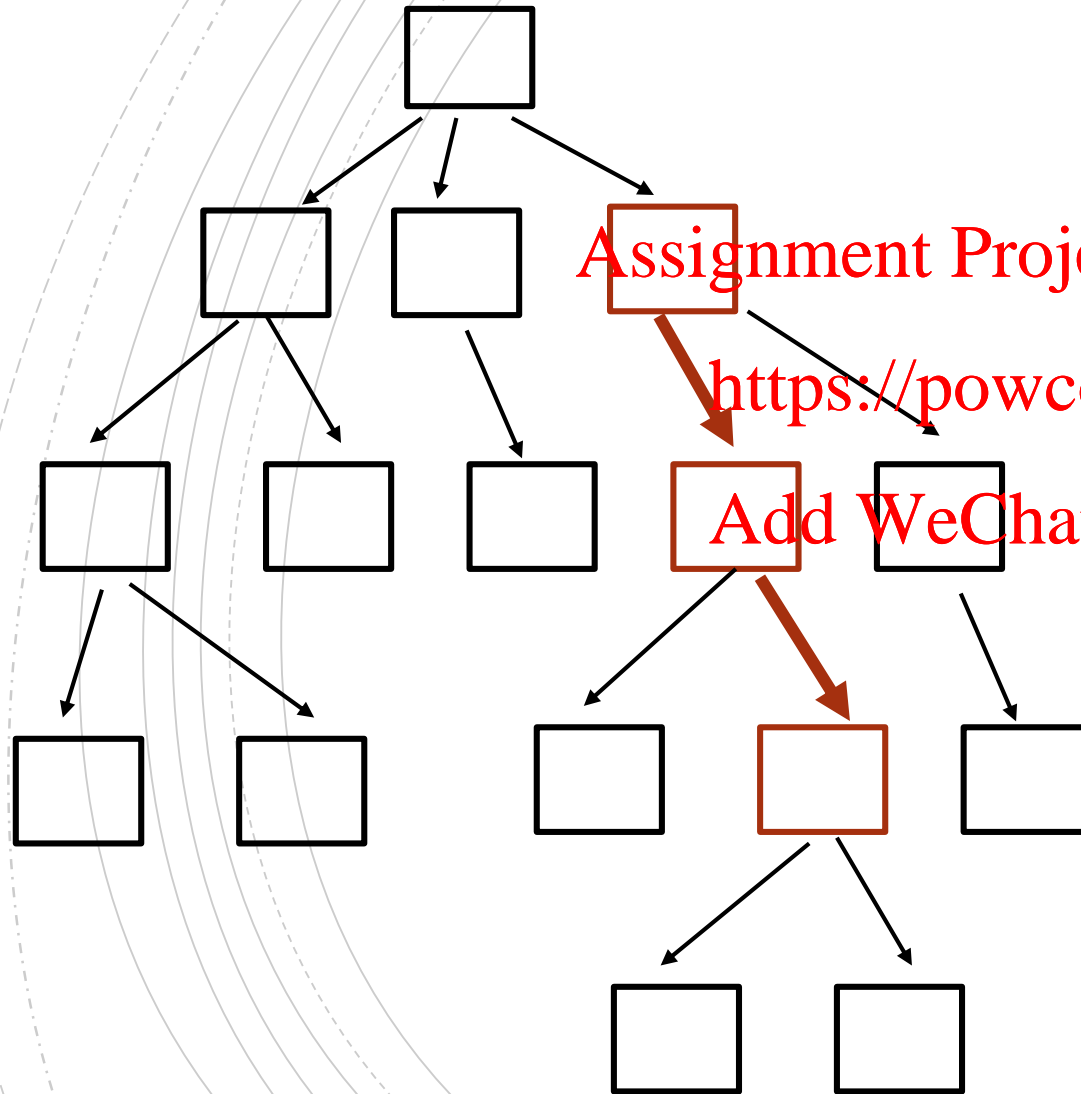
An *leaf (or external node)* is a node with no children.

leaves (external nodes)

(e.g. files or empty directories)



# TREE TERMINOLOGY



A *path* in a tree is a sequence of nodes  $(v_1, v_2, \dots, v_k)$  such that  $(v_i, v_{i+1})$  is an edge.

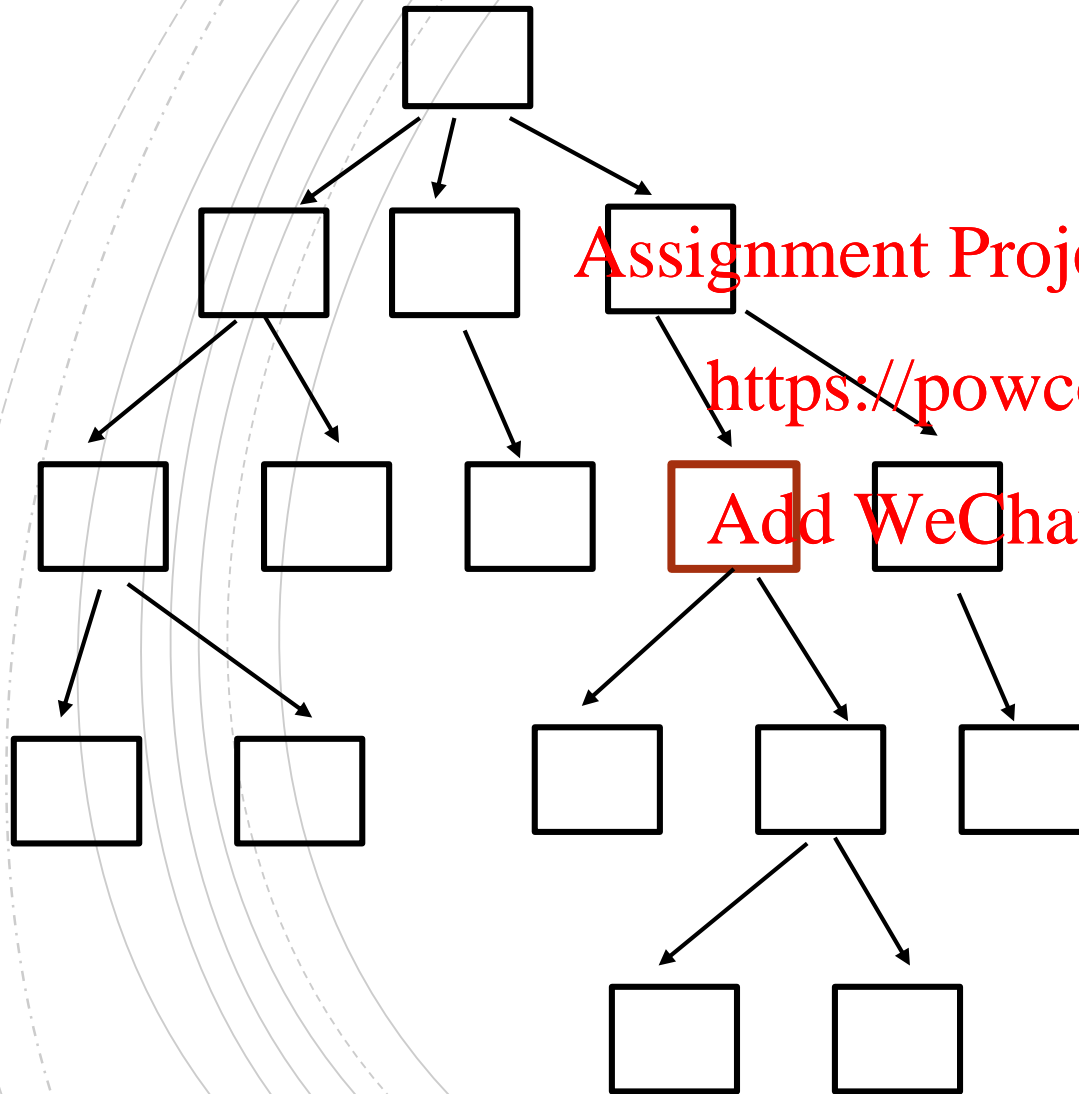
Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

The *length* of a path is the number of edges in the path (number of nodes in the path – 1)

# TREE TERMINOLOGY



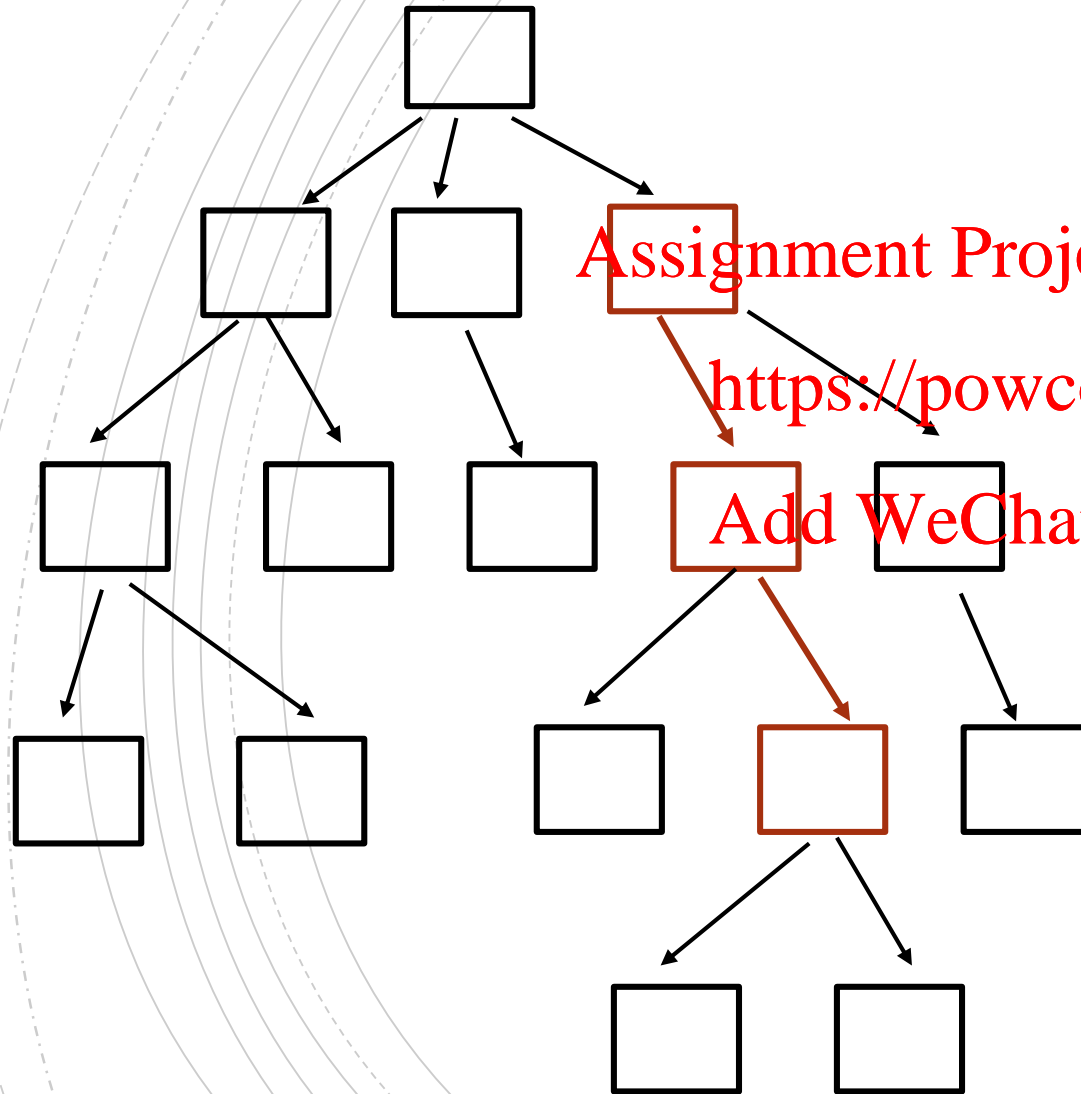
Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

A path with just one node ( $v_1$ ) has length = 0, since it has no edges.

## QUICK QUESTION



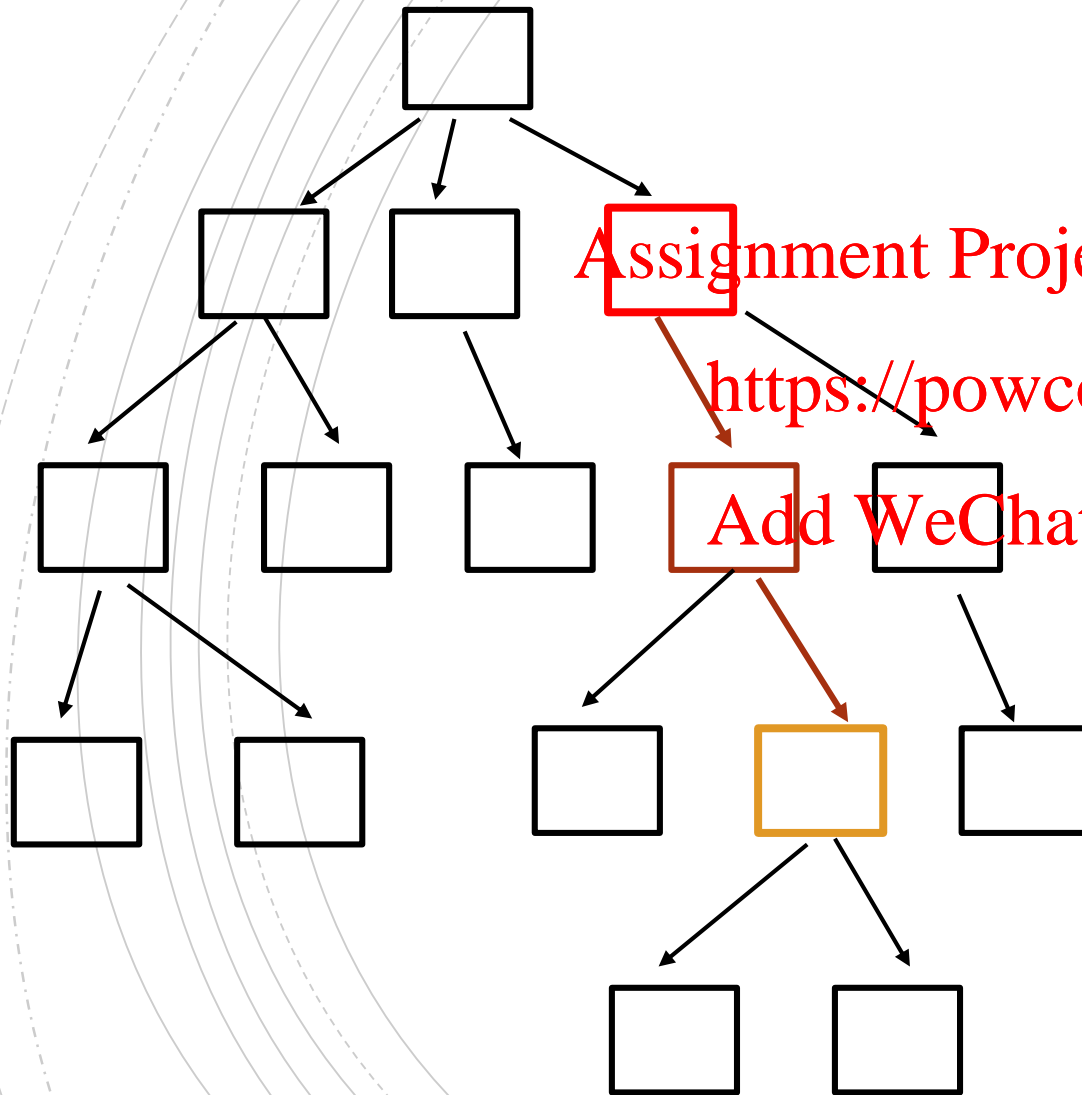
Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

What is the path length?

# TREE TERMINOLOGY



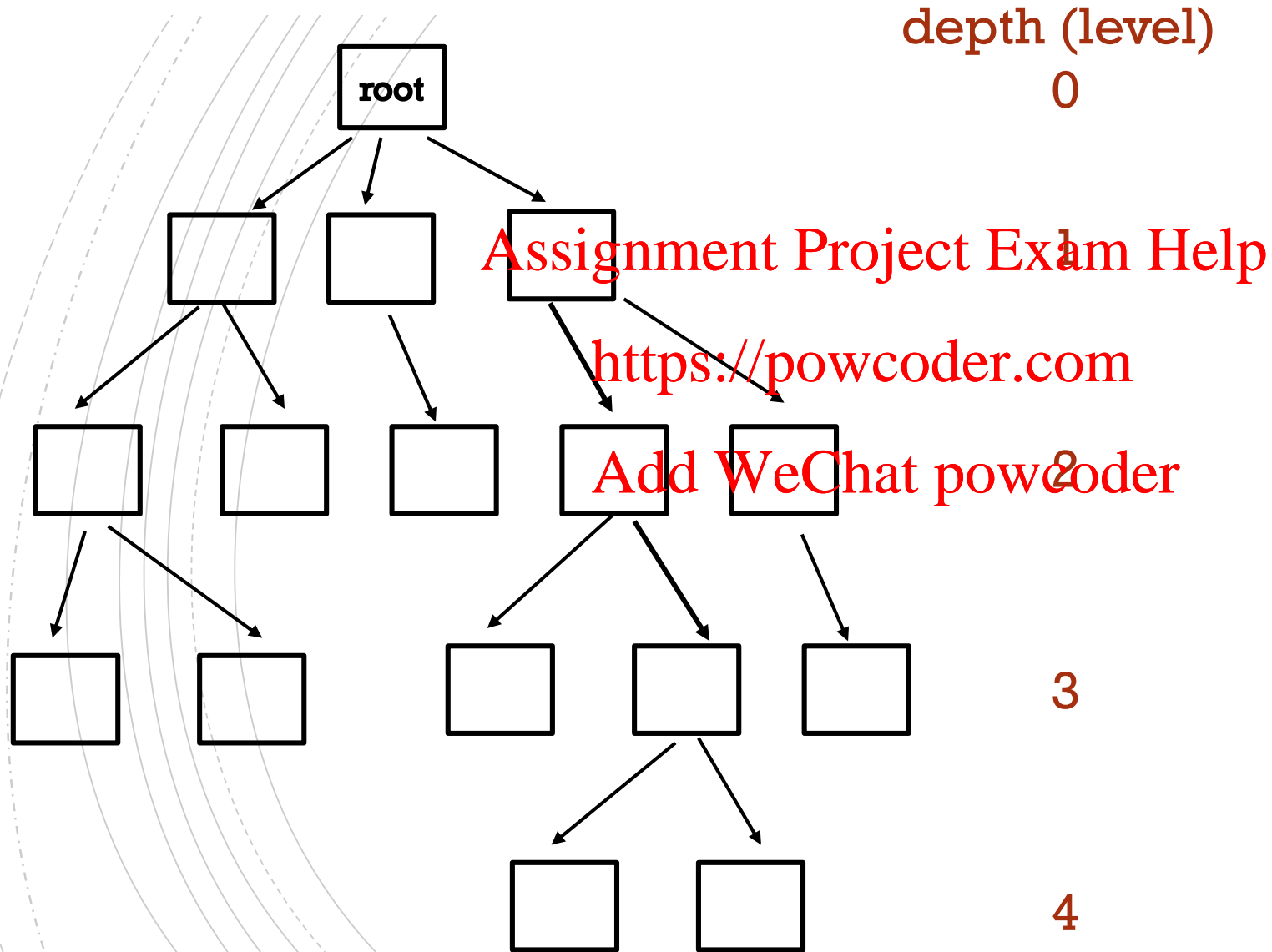
# Assignment Project Exam Help

~~<https://powcoder.com>~~

# Add WeChat powcoder

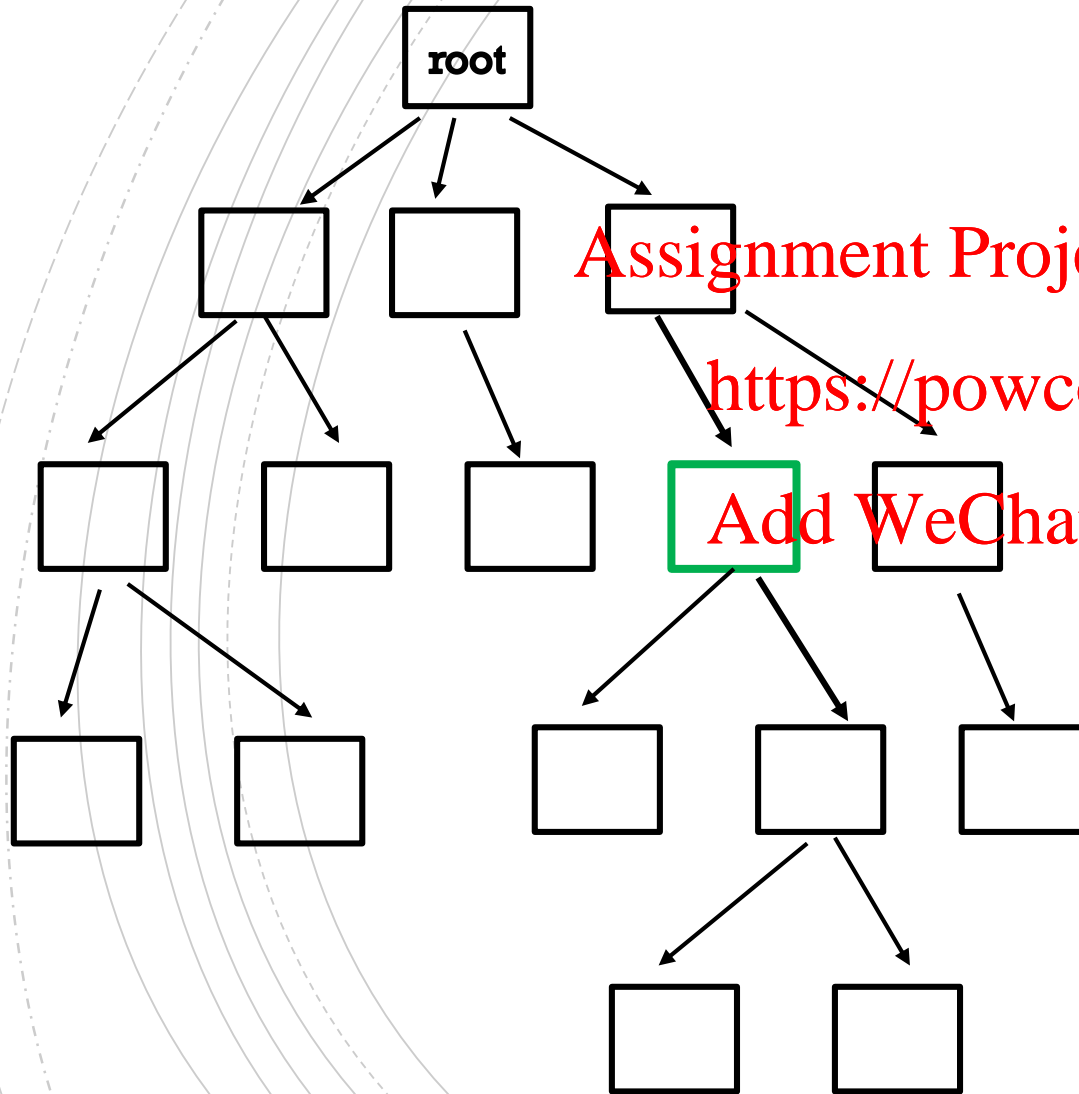
Node  $v$  is an *ancestor* of node  $w$  if there is a path from  $v$  to  $w$ .  
In such case, we also say that  $w$  is a *descendent* of node  $v$ .

## (ROOTED) TREE TERMINOLOGY



The *depth* (or *level*) of a node is the length of the path *from the root to the node*.

## (ROOTED) TREE TERMINOLOGY



Assignment Project Exam Help

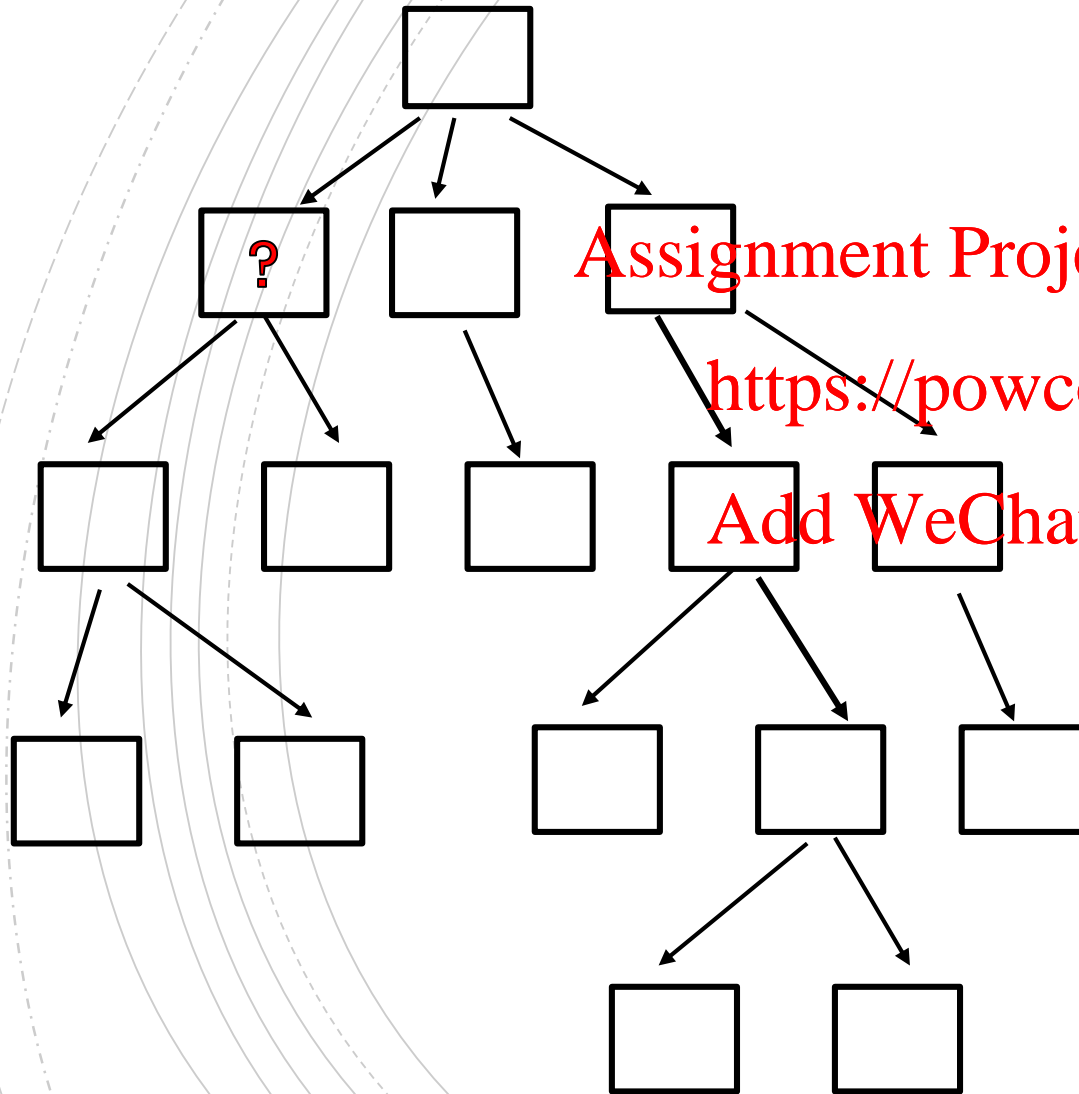
<https://powcoder.com>

Add WeChat powcoder

How can we compute the depth of a node  $v$ ?



## (ROOTED) TREE TERMINOLOGY



Assignment Project Exam Help

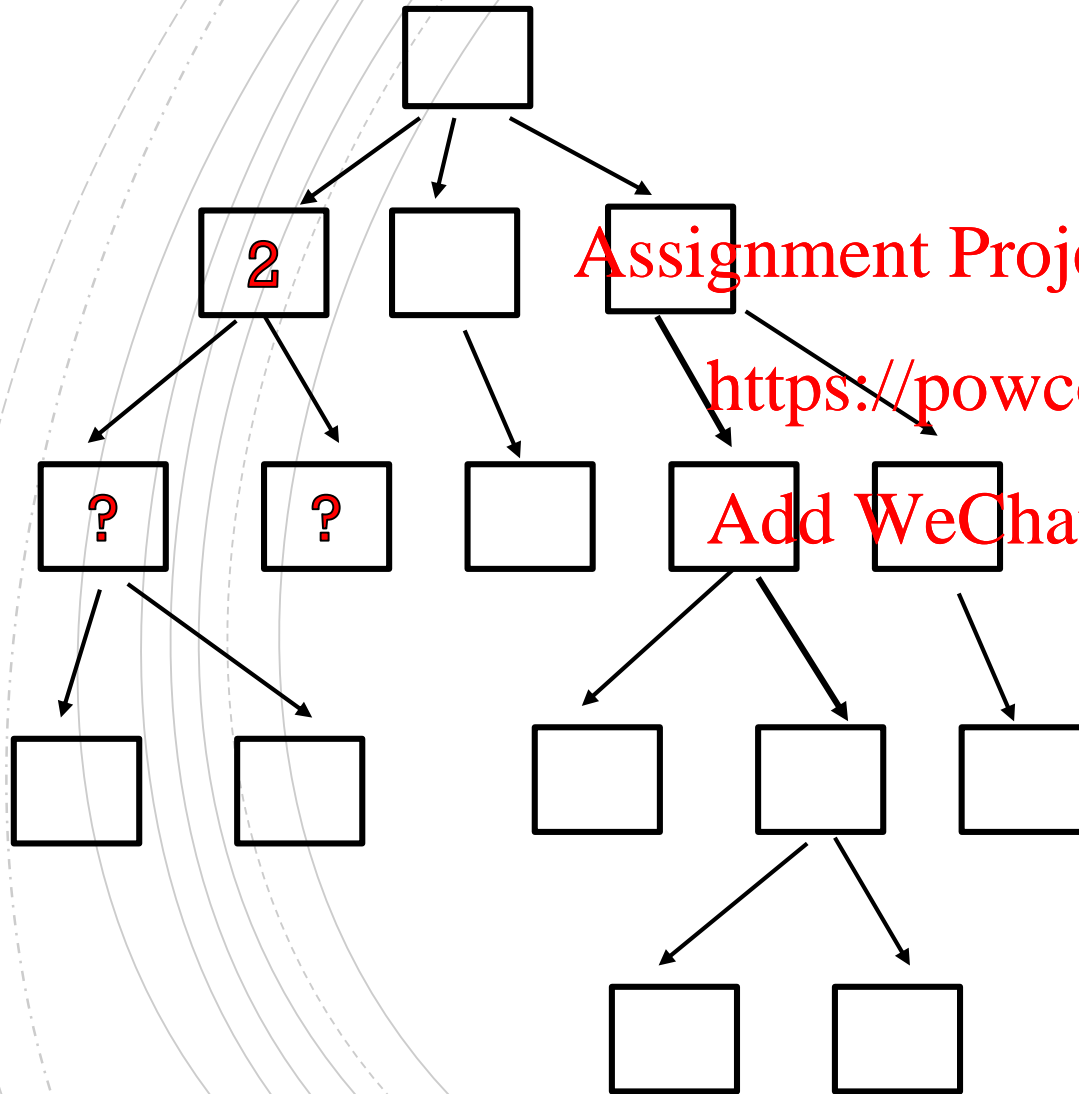
<https://powcoder.com>

Add WeChat powcoder

The *height* of a node is the maximum length of a path from that node to a leaf.



# TREE TERMINOLOGY



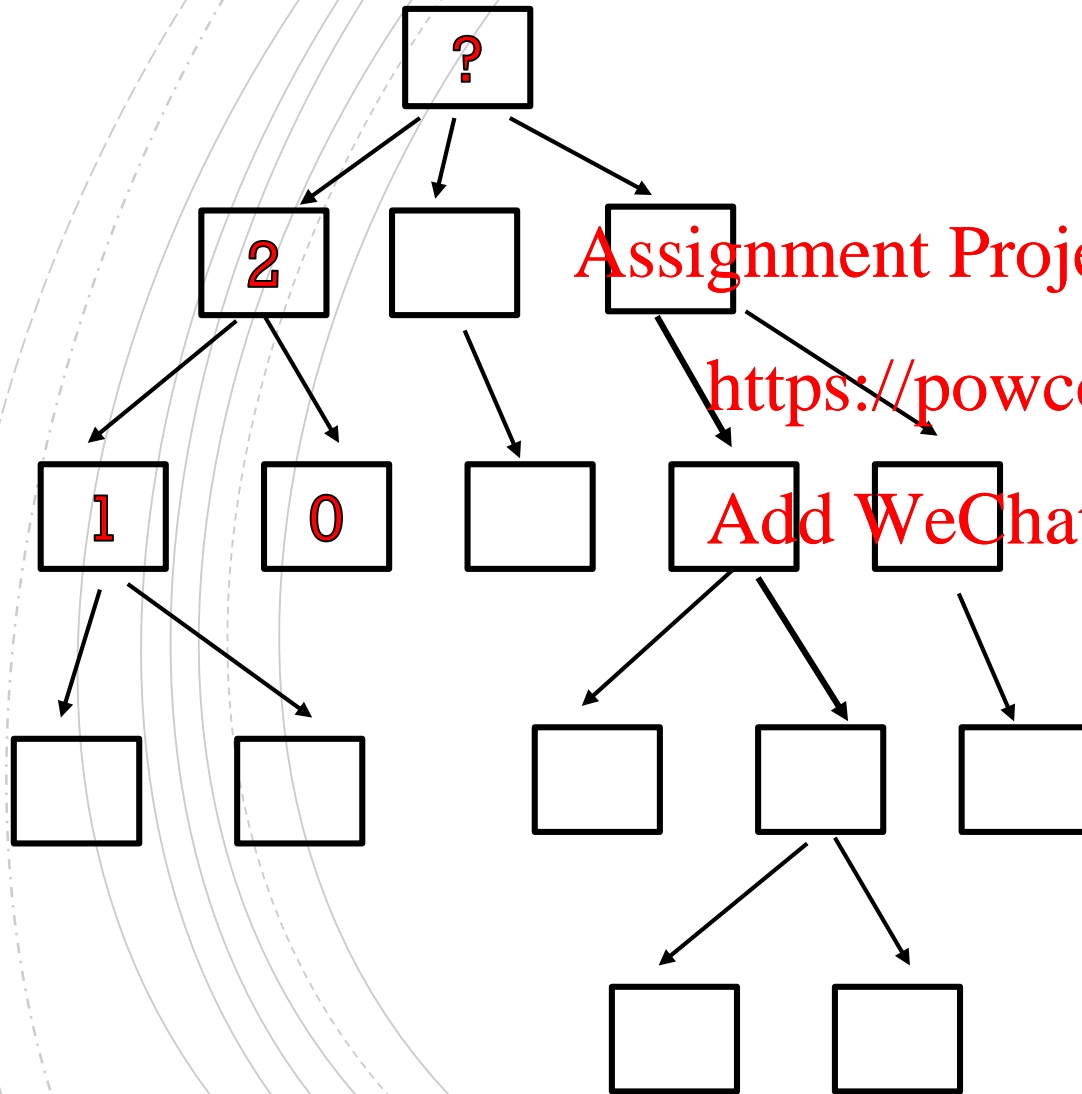
Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

The *height* of a node is the maximum length of a path from that node to a leaf.

# TREE TERMINOLOGY



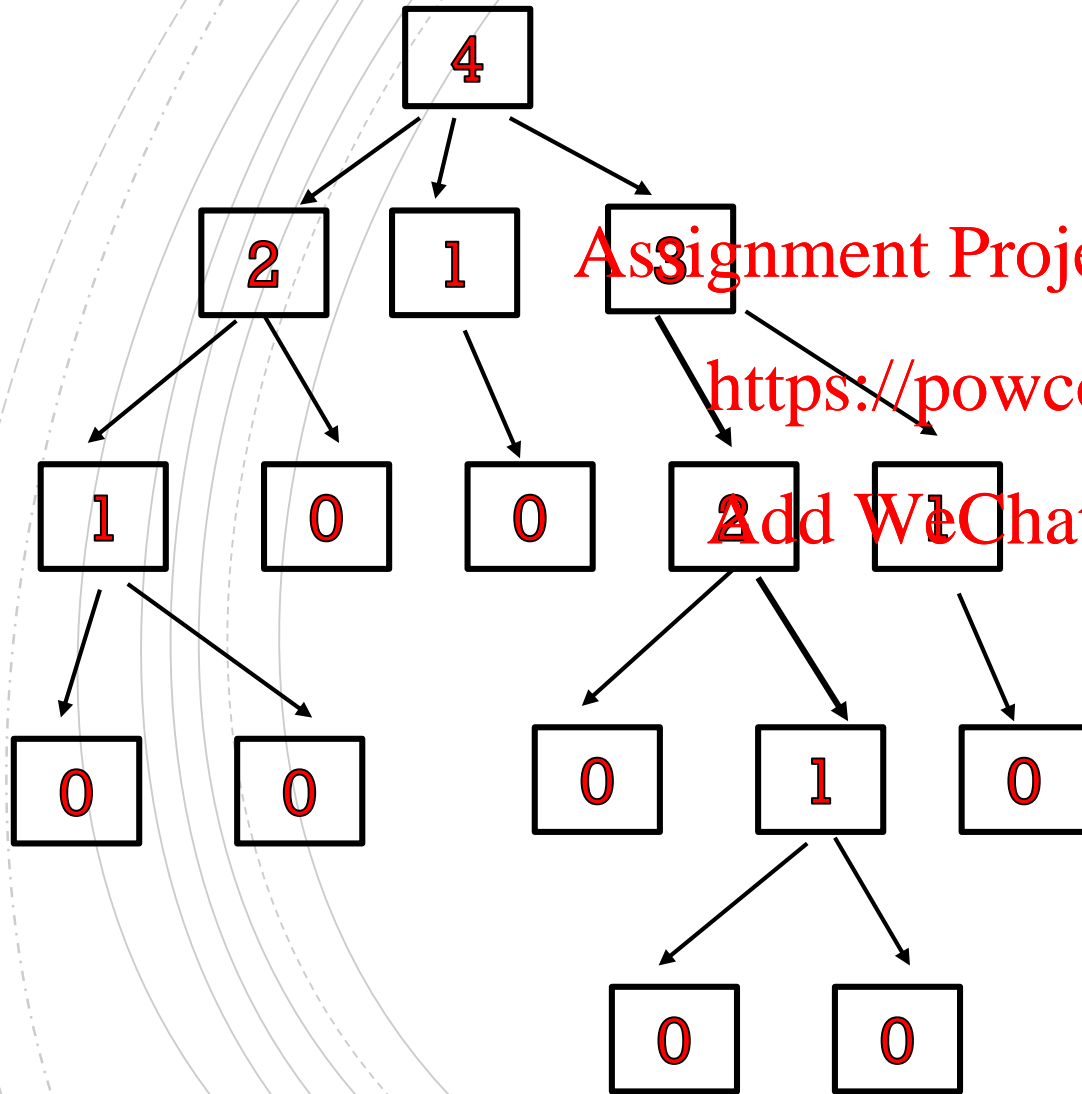
# Assignment Project Exam Help

~~<https://powcoder.com>~~

Add WeChat powcoder

The *height* of a node is the maximum length of a path from that node to a leaf.

# TREE TERMINOLOGY



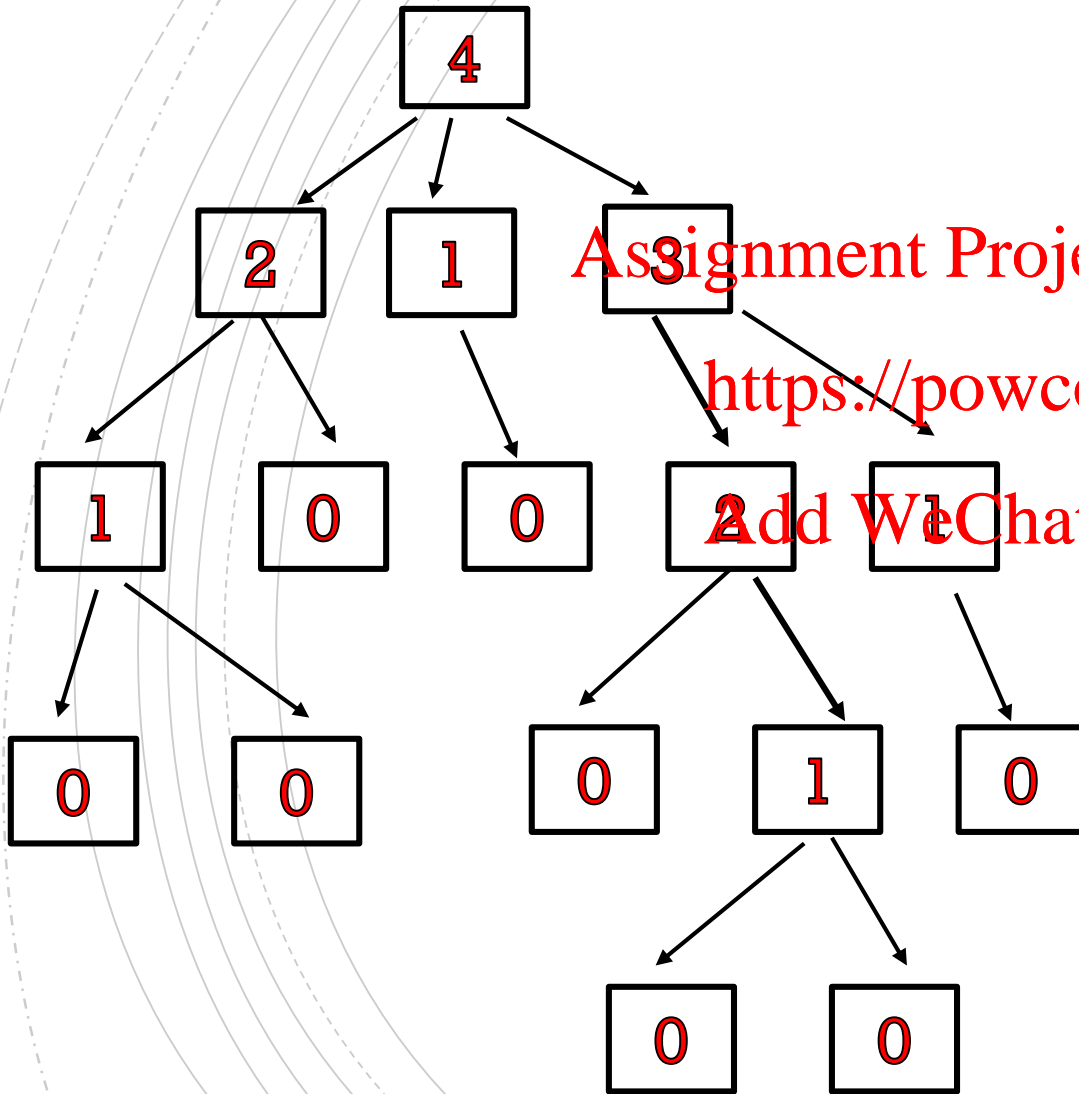
# Assignment Project Exam Help

~~<https://powcoder.com>~~

# Add WeChat powcoder

How can we compute the height of a node  $v$ ?

height(v)



# Assignment Project Exam Help

~~<https://powcoder.com>~~

## Add WeChat powcoder

```
height(v) {
    if (v is a leaf)
        return 0
    else {
        h = 0
        for each child w of v
            h = max(h, height(w))
        return 1 + h
    }
}
```

Assignment Project Exam Help  
**IMPLEMENTATIONS**

<https://powcoder.com>

Add WeChat powcoder

# HOW TO IMPLEMENT A TREE IN JAVA?

Same idea as with linked lists:

- Create a data type to represent tree nodes.
- Represent a tree with a pointer to the root node.

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

```
class TreeNode<T> {  
    T element;  
  
    }  
}
```

# HOW TO IMPLEMENT A TREE IN JAVA?

Same idea as with linked lists:

- Create a data type to represent tree nodes.
- Represent a tree with a pointer to the root node.

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

```
class TreeNode<T> {  
    T element;  
    ArrayList<TreeNode<T>> children;  
  
    TreeNode<T> parent; // optional  
}
```

# HOW TO IMPLEMENT A TREE IN JAVA?

Same idea as with linked lists:

- Create a data type to represent tree nodes.
- Represent a tree with a pointer to the root node.

Assignment Project Exam Help

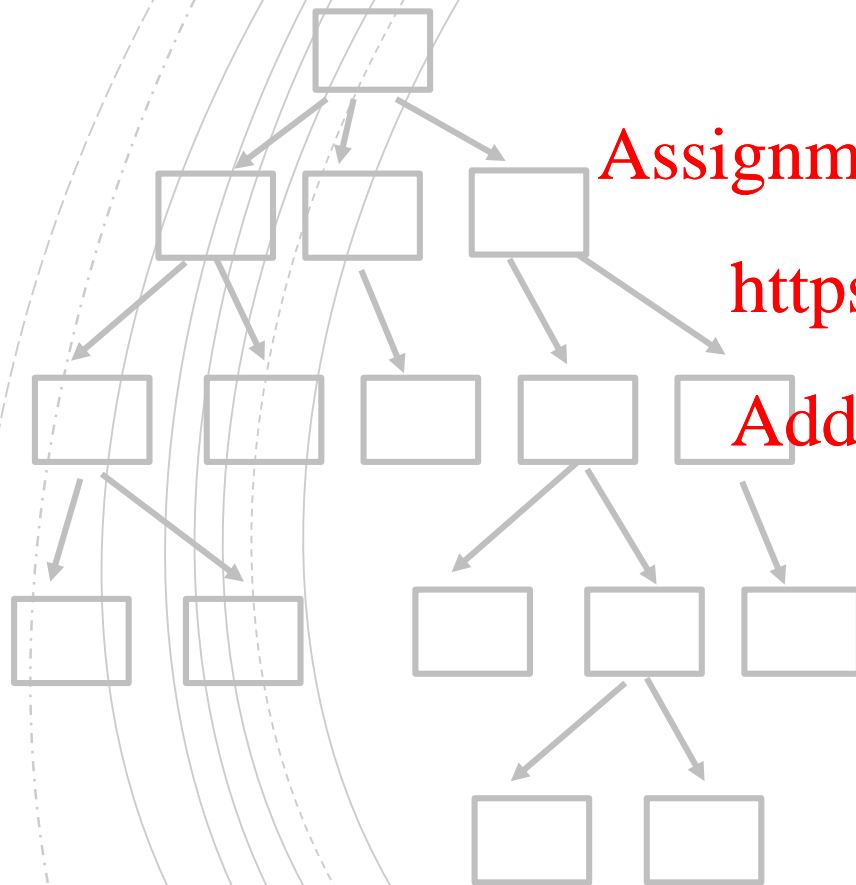
<https://powcoder.com>

Add WeChat powcoder

```
class Tree<T>{  
    TreeNode<T> root;  
:  
}  
  
class TreeNode<T>{  
    T element;  
    ArrayList<TreeNode<T>> children;  
    TreeNode<T> parent; // optional  
}  
}
```



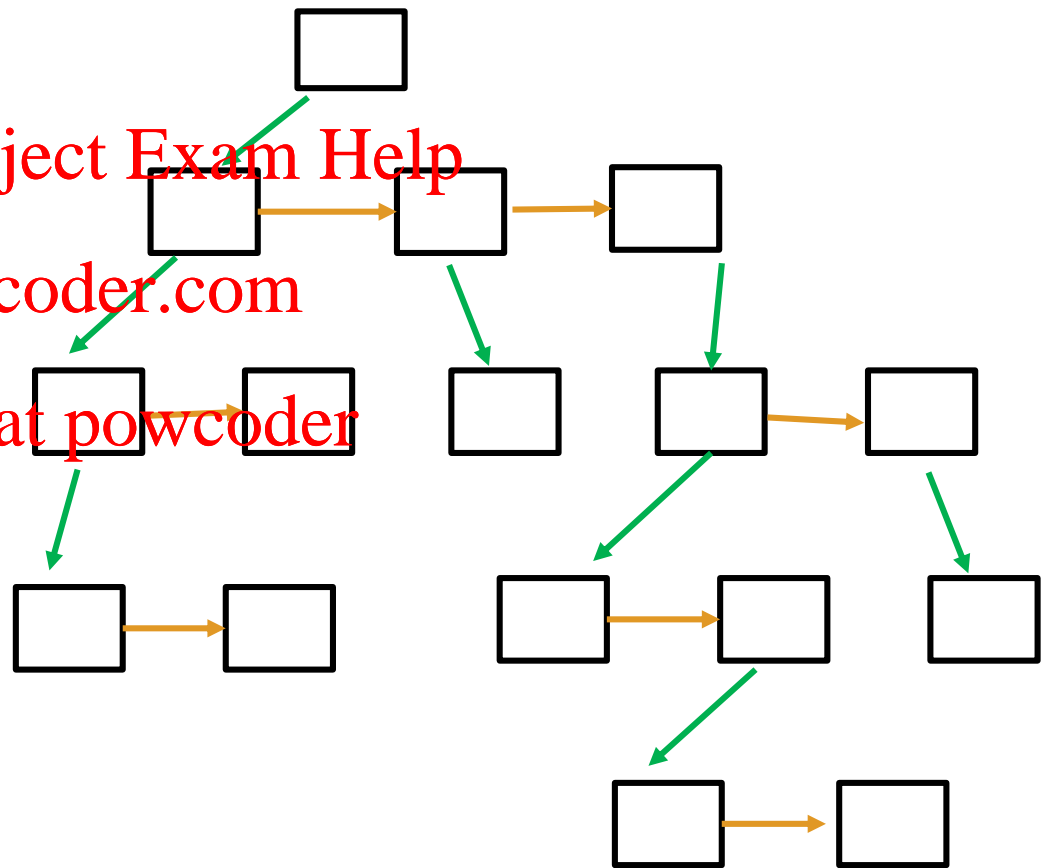
## ANOTHER COMMON IMPLEMENTATION: 'first child, next sibling'



# Assignment Project Exam Help

<https://powcoder.com>

# Add WeChat powcoder



## ANOTHER COMMON IMPLEMENTATION: 'first child, next sibling'

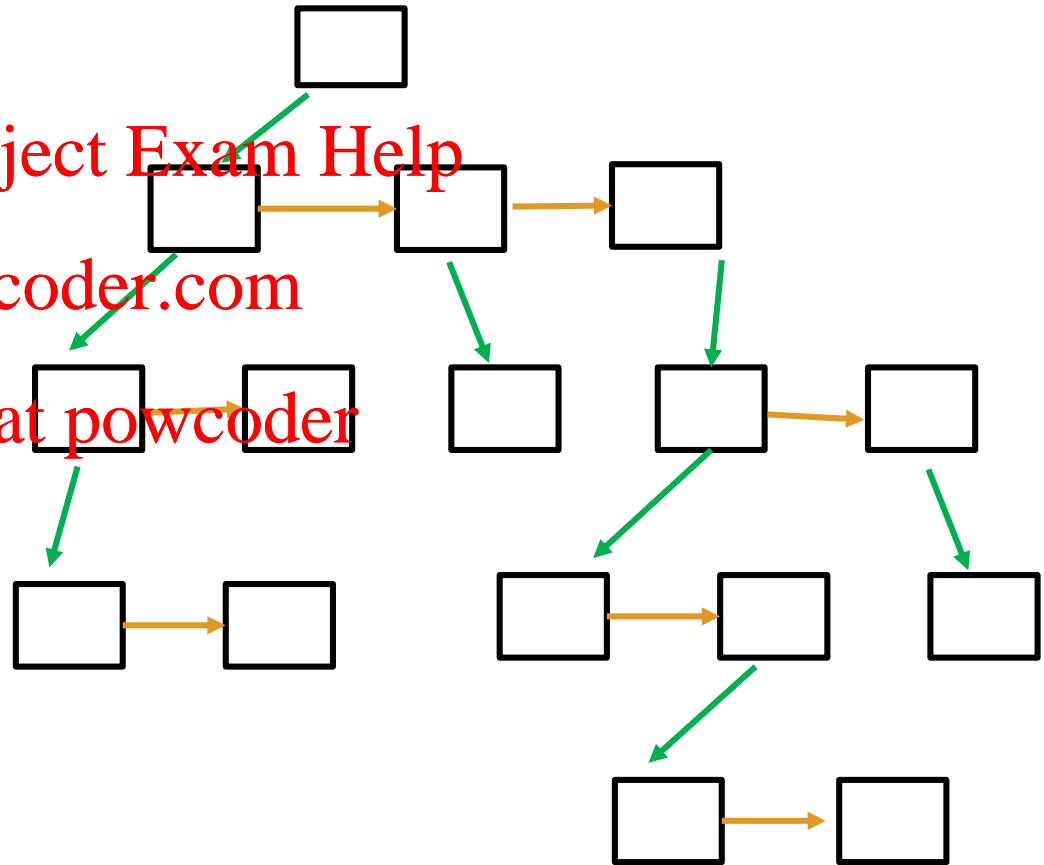
(similar to singly linked lists)

```
class Tree<T>{
    TreeNode<T> root;
    :
    class TreeNode<T>{
        T element;
        TreeNode<T> firstChild;
        TreeNode<T> nextSibling;
        :
    }
}
```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder



## ANOTHER COMMON IMPLEMENTATION: 'first child, next sibling'

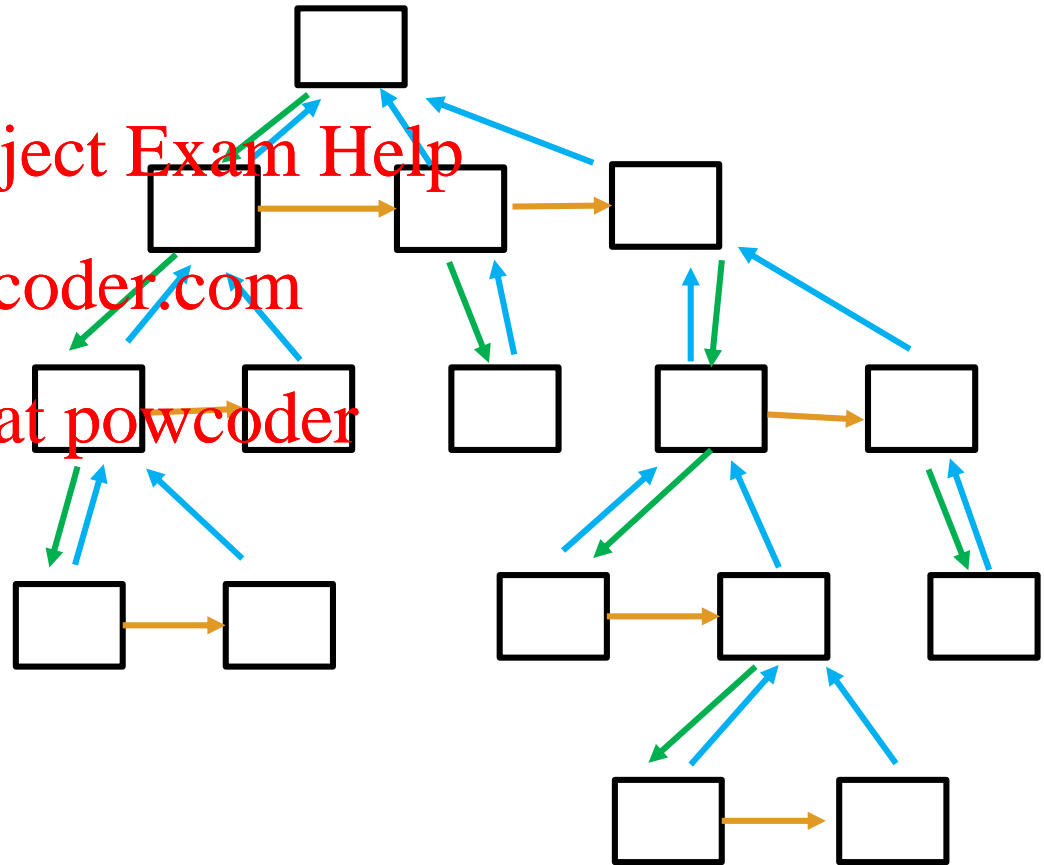
(similar to singly linked lists)

```
class Tree<T>{
    TreeNode<T> root;
    :
    class TreeNode<T>{
        T element;
        TreeNode<T> firstChild;
        TreeNode<T> nextSibling;
        TreeNode<T> parent;
    }
}
```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder



# A TREE OF WHAT? EACH NODE HAS AN ELEMENT!

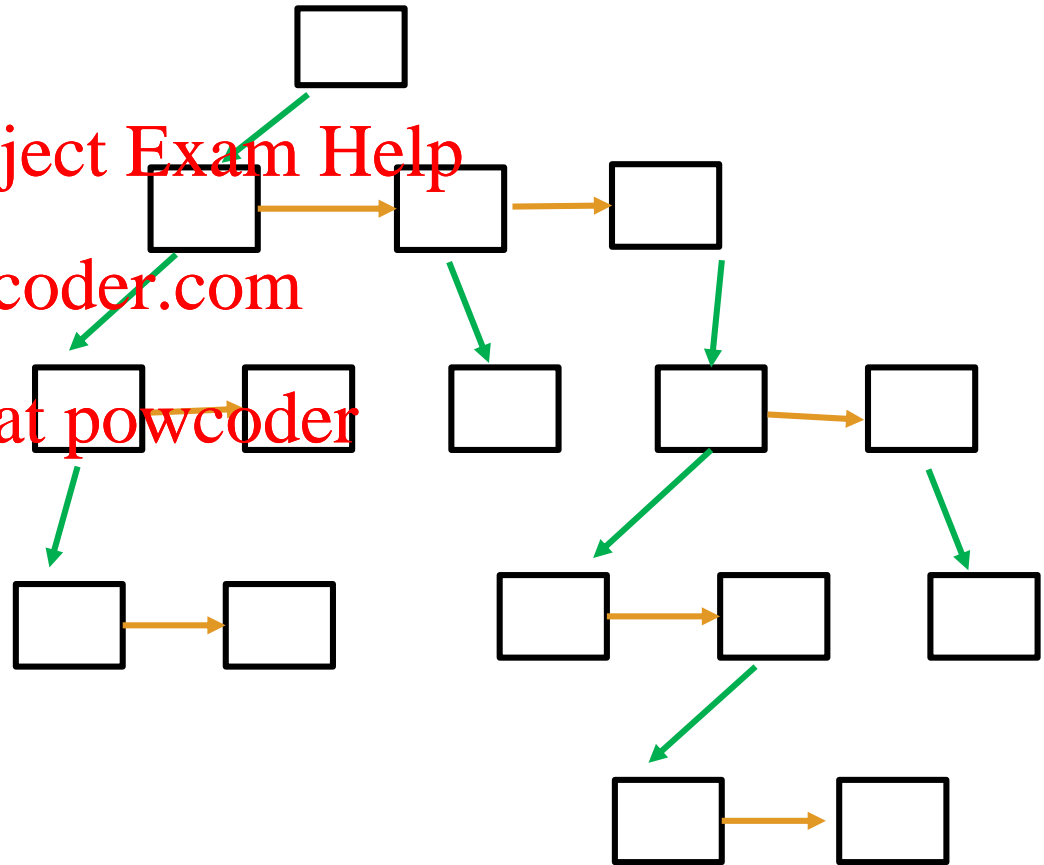
(NOT ILLUSTRATED ON THE RIGHT)

```
class Tree<T>{
    TreeNode<T> root;
    :
    class TreeNode<T>{
        T element;
        TreeNode<T> firstChild;
        TreeNode<T> nextSibling;
        :
    }
}
```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder



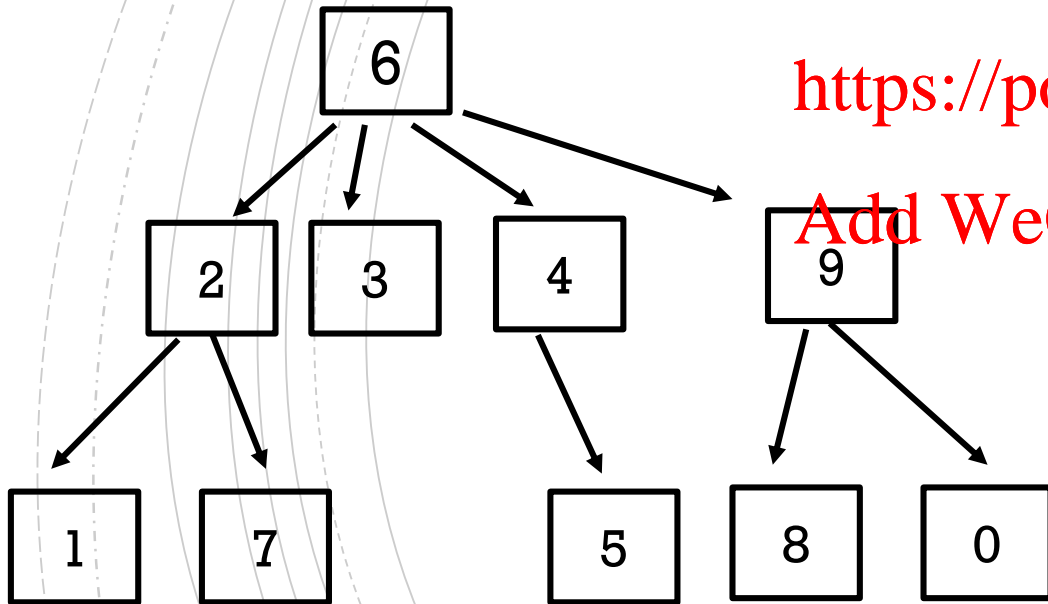
## ANOTHER EXERCISES

Write this tree using the first child, next sibling representation.

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder



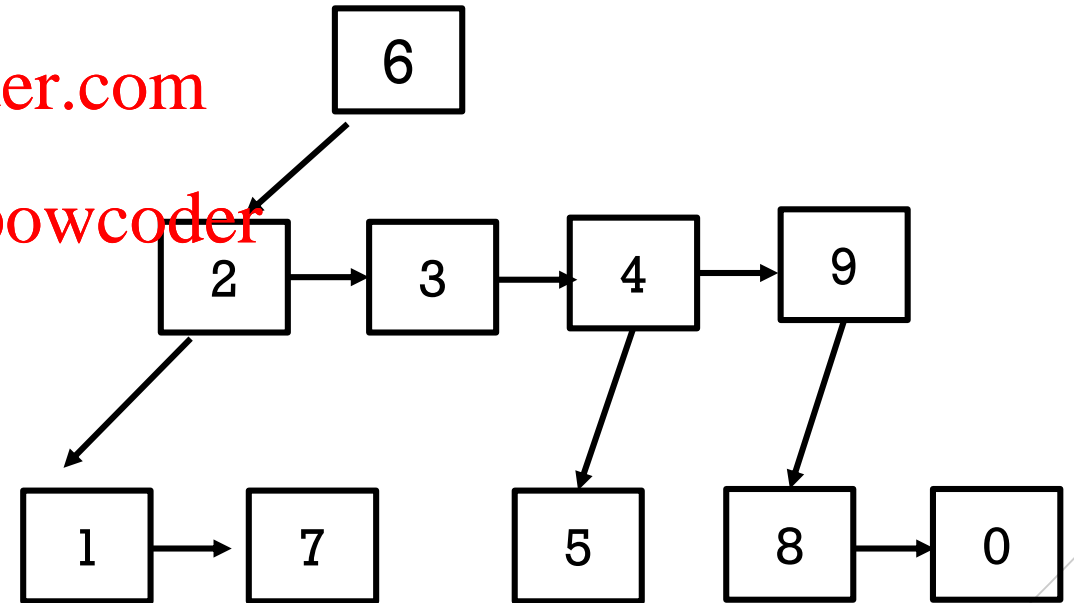
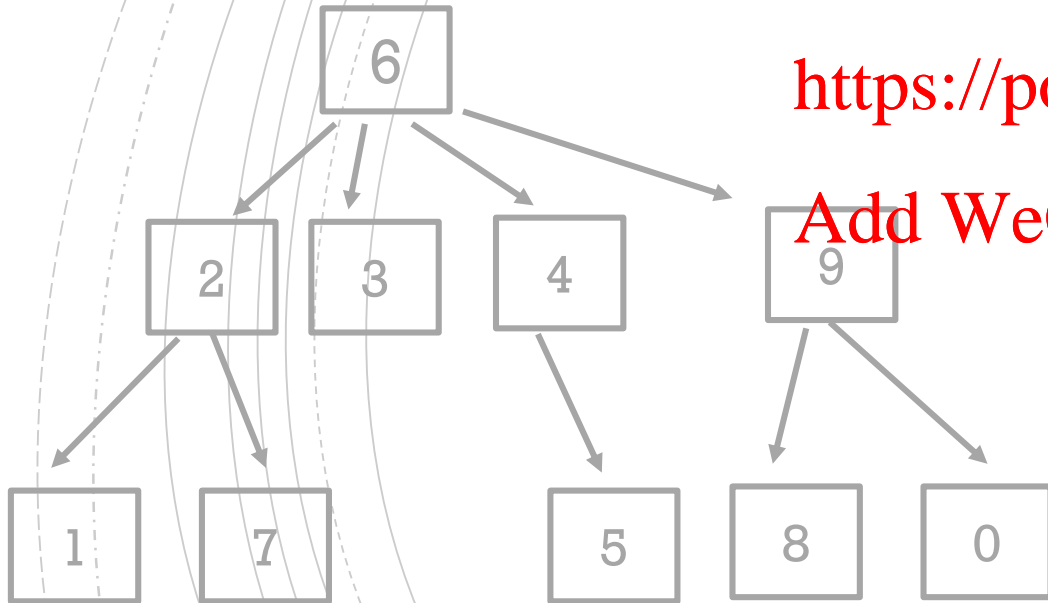
## SOLUTION

Write this tree using the first child, next sibling representation.

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder





# Coming Soon

**Assignment Project Exam Help**

**In the next video:**

■ **<https://powcoder.com>**  
**Tree Traversals**

**Add WeChat powcoder**