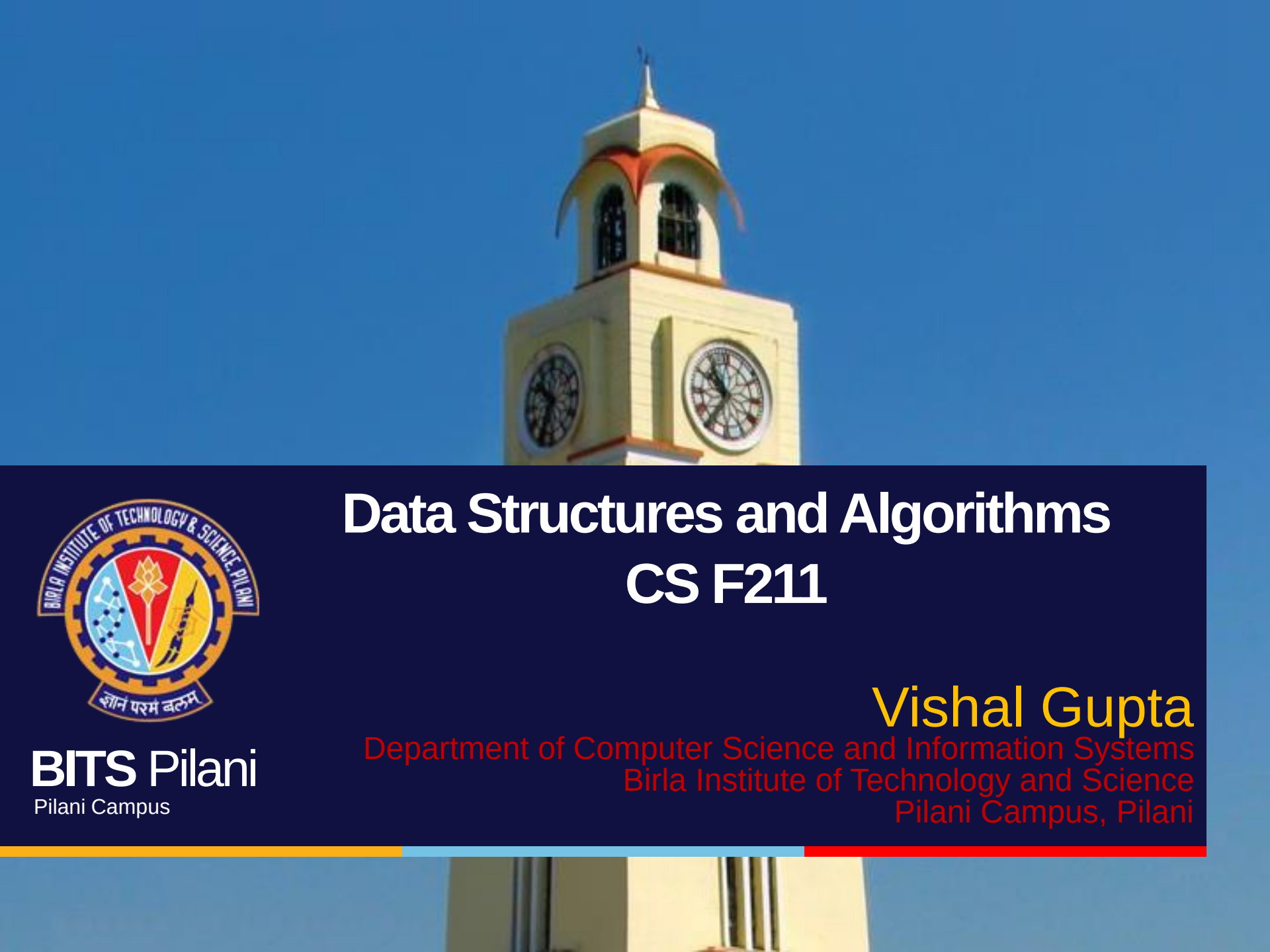


Data Structures and Algorithms
CS F211

Vishal Gupta

Department of Computer Science and Information Systems
Birla Institute of Technology and Science
Pilani Campus, Pilani

BITS Pilani
Pilani Campus

Non-linear Data Structures: Trees

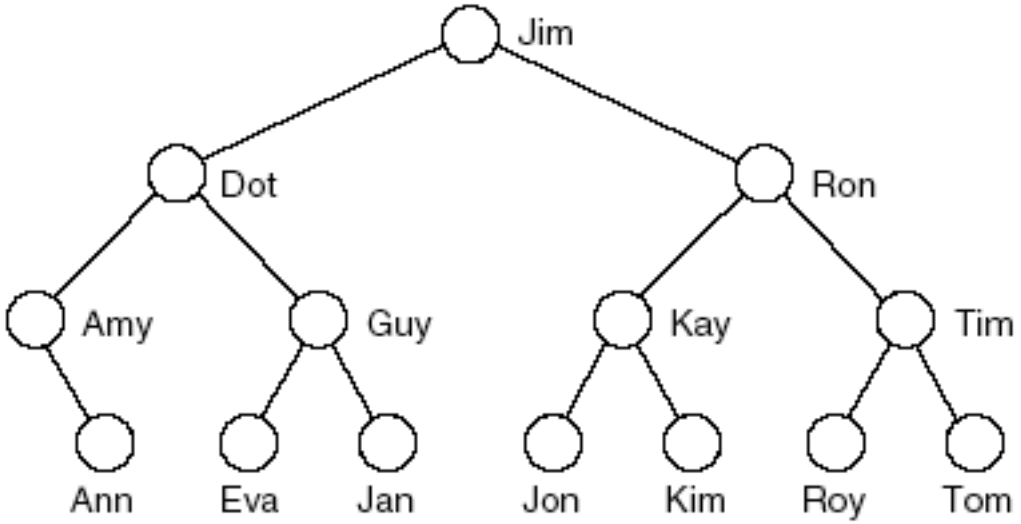
Tree

innovate achieve lead

Tree represents hierarchy.

Examples of trees:

- Directory tree
- Family tree
- Company organization c
-Table of contents

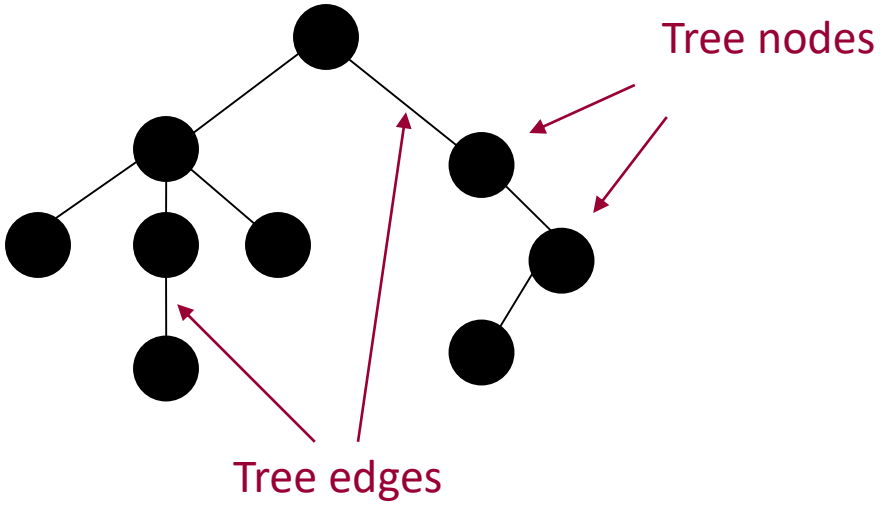


-structure resembles branches of a “tree”, hence the name.

Trees

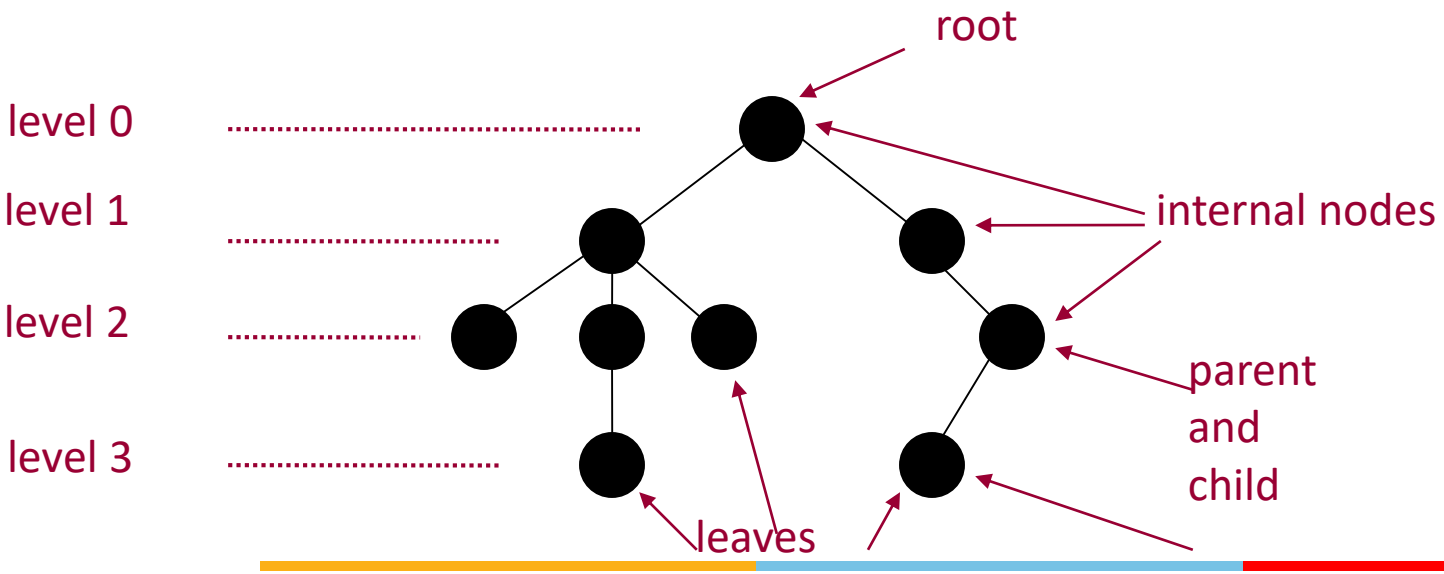
Trees have **nodes**. They also have **edges** that connect the nodes.

- Between two nodes there is *always only one* path.



Trees: More Definitions

- Trees that we consider are rooted. Once the **root** is defined (by the user) all nodes have a specific **level**.
- Trees have **internal** nodes and **leaves**. Every node (except the root) has a **parent** and it also has zero or more **children**.



Tree Terminology (1)

- A vertex (or node) is an object that can have a name and can carry other associated information.
- The first or top node in a tree is called the root node.
- An edge is a connection between two vertices.
- A path in a tree is a list of distinct vertices in which successive vertices are connected by edges in the tree.
- The defining property of a tree is that there is precisely one path connecting any two nodes.
- A disjoint set of trees is called a forest.
- Nodes with no children are leaves, terminal or external nodes.

Tree Terminology (2)

Child of a node u :- Any node reachable from u by 1 edge.

Parent node :- If b is a child of a, then a is the parent of b.

- All nodes except root have exactly one parent.

Subtree:-any node of a tree, with all of its descendants.

Depth of a node :

- Depth of root node is 0.

-Depth of any other node is 1 greater than depth of its parent.

Tree Terminology (3)

The size of a tree is the number of nodes in it

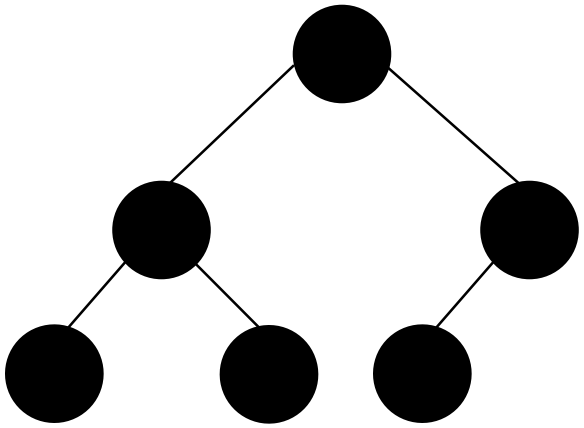
Height : ***Maximum of all depths.***

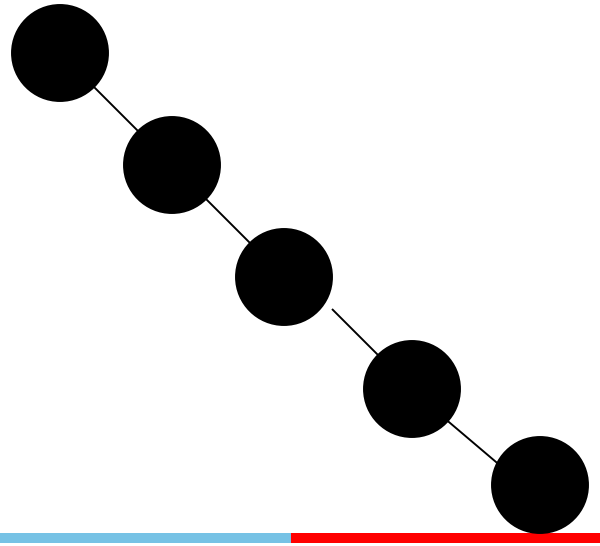
Each node except the root has exactly one node above it in the tree, (i.e. it's parent), and we extend the family analogy talking of children, siblings, or grandparents.
Nodes that share parents are called **siblings.**

Binary Trees

Definition: A binary tree is either empty or it consists of a root together with two binary trees called the left subtree and the right subtree.

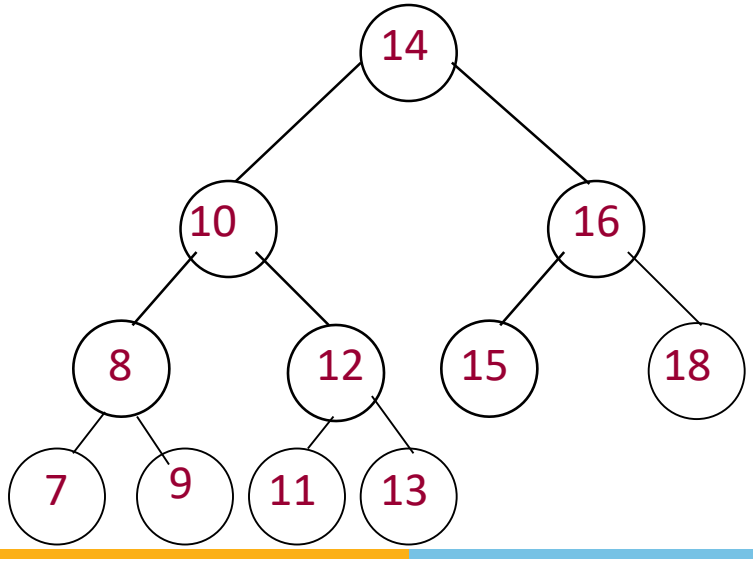
A **binary tree** is a tree in which each node has 2 children.





Complete Binary Trees

- Nodes in trees can contain **keys** (letters, numbers, etc)
- **Complete binary tree**: A binary tree in which every level, except possibly the deepest, is completely filled. At depth n, the height of the tree, all nodes must be as far left as possible.

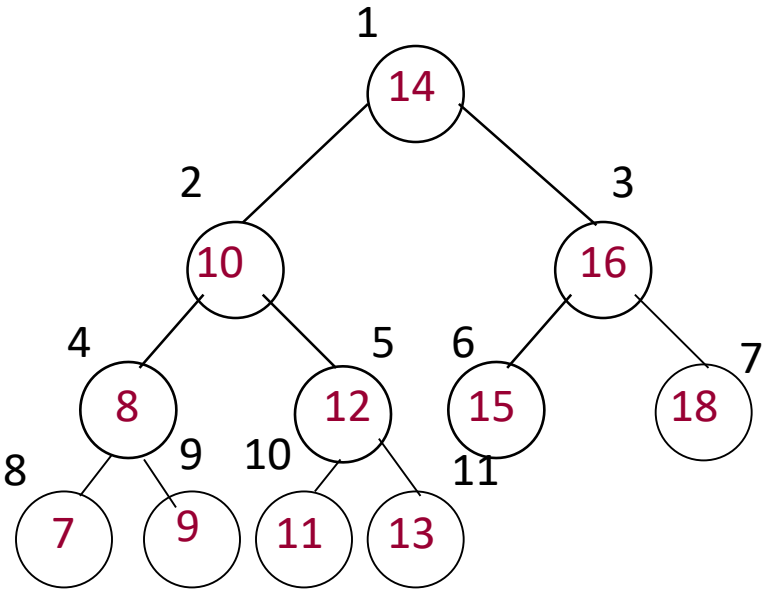


Complete Binary Trees: Array Representation

Complete Binary Trees can be represented in memory with the use of an array A so that all nodes can be accessed in O(1) time:

- Label nodes sequentially top-to-bottom and left-to-right
- Left child of A[i] is at position A[2i]
- Right child of A[i] is at position A[2i + 1]
- Parent of A[i] is at A[i/2]

Complete Binary Trees: Array Representation

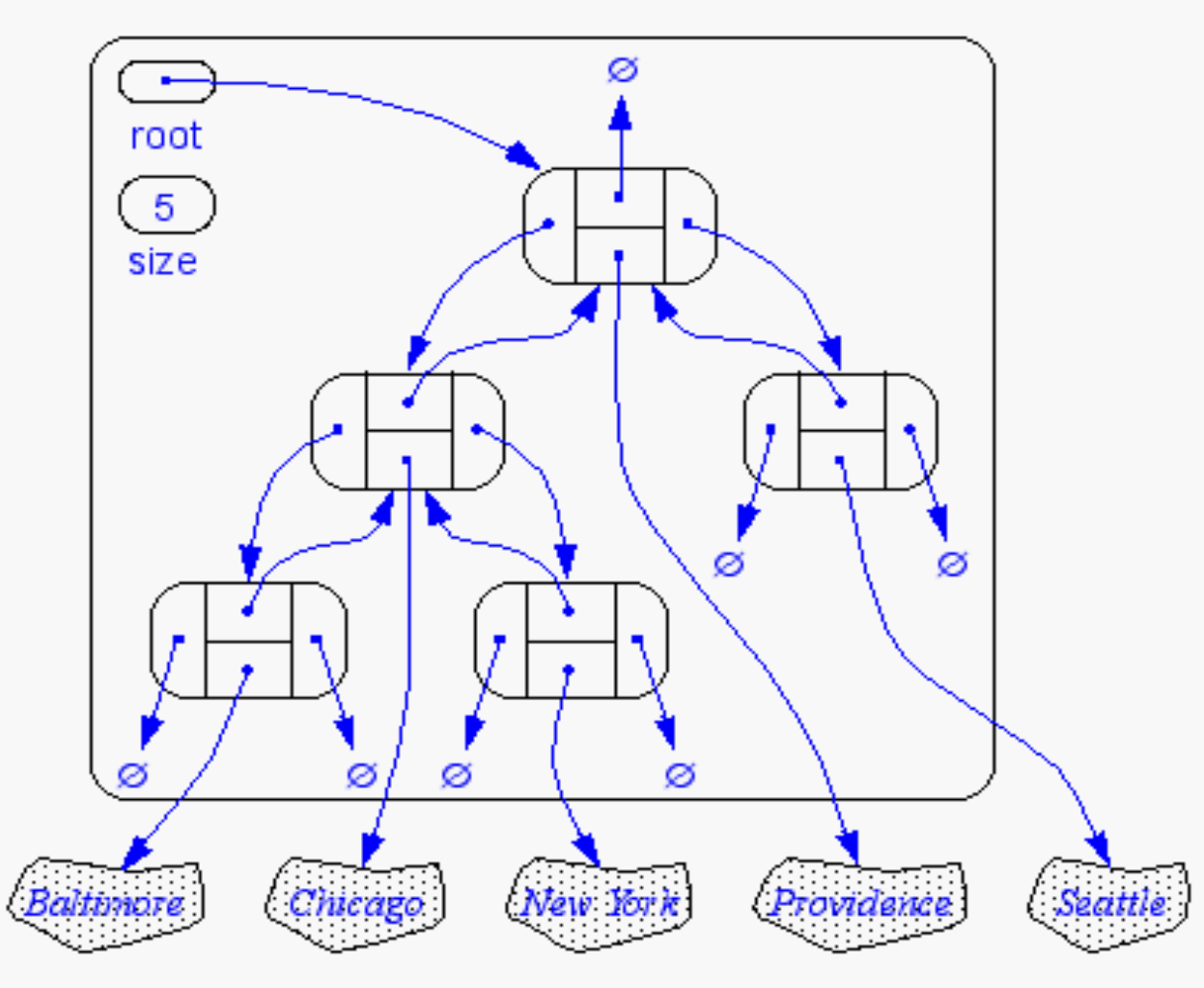


Binary Trees: Linked List Representation

A **Binary tree** is a linked data structure. Each node contains data (including a key and satellite data), and pointers left, right and p.

- ***Left* points to the left child of the node.**
- ***Right* points to the right child of the node.**
- ***p* points to the parent of the node.**
- If a child is missing, the pointer is NIL.
- If a parent is missing, p is NIL.
- The **root** of the tree is the only node for which p is NIL.
- Nodes for which both left and right are NIL are **leaves.**

Binary Trees: Linked List Representation



Height, Depth, and Level

- *The height of a node is the number of edges on the longest downward path between that node and a leaf.*
- *The depth of a node is the number of edges from the node to the tree's root node.*
- Level and depth are same. Although, some textbooks say that level = depth + 1

Problem - 1

1. Draw a Binary tree with height 7 and maximum number of nodes.

Problem - 1

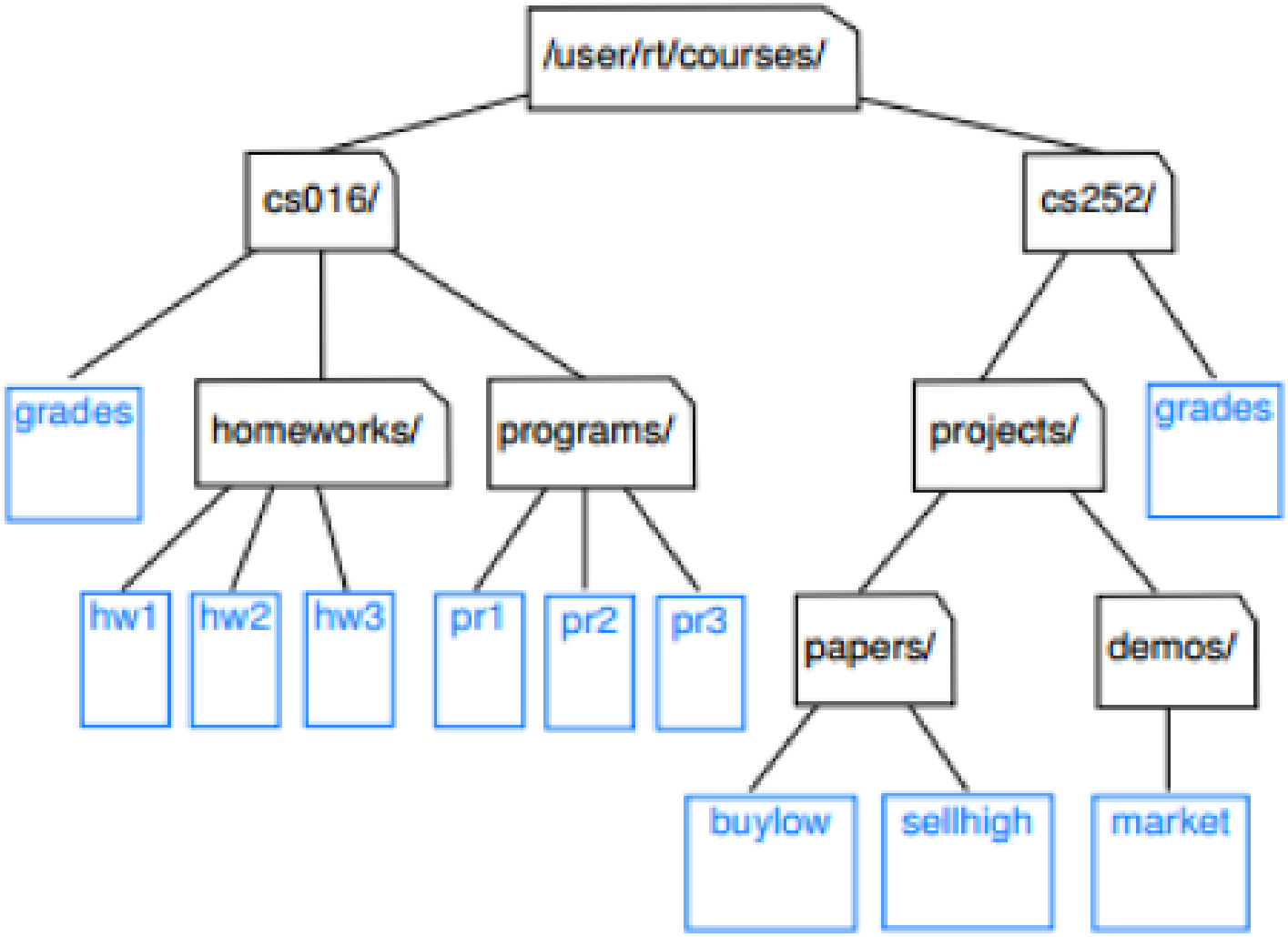
2. Is a complete binary tree balanced?

   Is vice versa also true?

Problem - 1

1. What is the minimum and maximum number of external nodes for a binary tree with height h? Justify your answer.

Problem


/user/rt/courses/
cs016/
cs252/
grades
homeworks/
programs/
projects/
grades
hw1
hw2
hw3
pr1
pr2
pr3
papers/
demos/
buylow
sellhigh
market

Problem 2

What are the **internal nodes**?

What are the **leaf nodes**?

How many **descendants** does node cs016/ have?

How many **ancestors** does node cs016/ have?

What are the **siblings** of node homeworks/?

Which nodes are in the **subtree** rooted at node projects/?

What is the **depth** of node papers/?

What is the **height** of the tree?

Problem

What does this algorithm computes ?

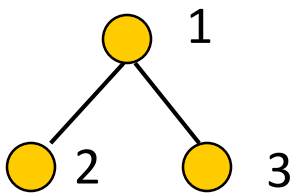
Algorithm 1

```
1: procedure E(Tree T, node v)
2:     if v is the root of T then
3:         return 0
4:     else
5:         return 1+ E(T, w) , where w is the parent of v in T
6:     end if
7: end procedure
```

Binary Tree Traversals

- A binary tree is defined recursively: it consists of a root, a left subtree and a right subtree
- To traverse (or walk) the binary tree is to visit each node in the binary tree exactly once.
- Tree traversals are naturally recursive.
- Since a binary tree has three parts, there are six possible ways to traverse the binary tree:
  - root, left, right : preorder (root, right, left)
  - left, root, right: inorder (right, root, left)
  - left, right, root: postorder (right, left, root)

Binary Tree Traversals



preorder: 1 2 3

inorder: 2 1 3

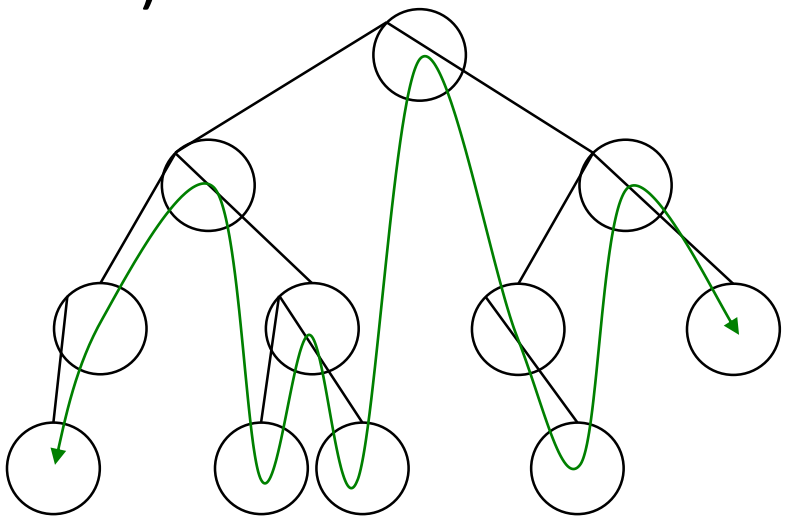
postorder: 2 3 1

Tree Traversal: InOrder

In-order traversal

- Visit left subtree (if there is one) In Order
- print the key of the current node
- Visit right subtree (if there is one) In Order

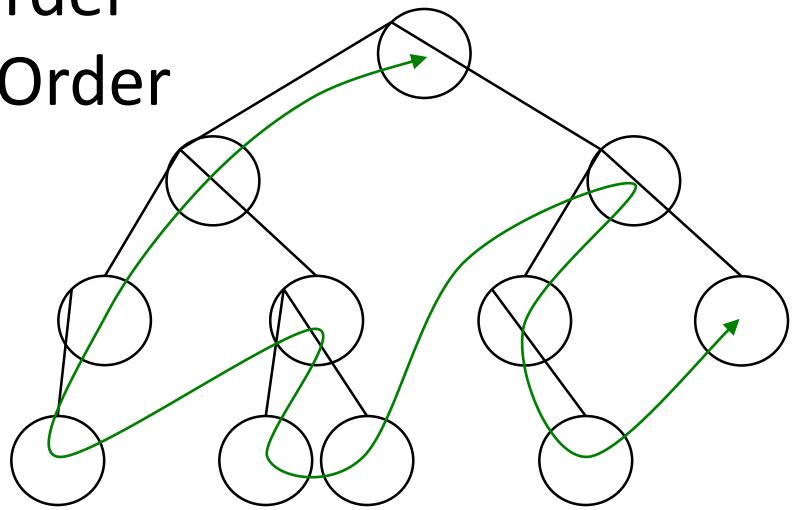
I



Tree Traversal: PreOrder

Another common traversal is **PreOrder**.
It goes as deep as possible (visiting as it goes) then left to right

- print root
- Visit left subtree in PreOrder
- Visit right subtree in PreOrder



Tree Traversal: PostOrder

PostOrder traversal also goes as deep as possible, but only visits internal nodes during backtracking.

recursive:

- Visit left subtree in PostOrder
- Visit right subtree in PostOrder
- print root