Trees

(Part 1, Theoretical)

CSE 2320 – Algorithms and Data Structures
University of Texas at Arlington

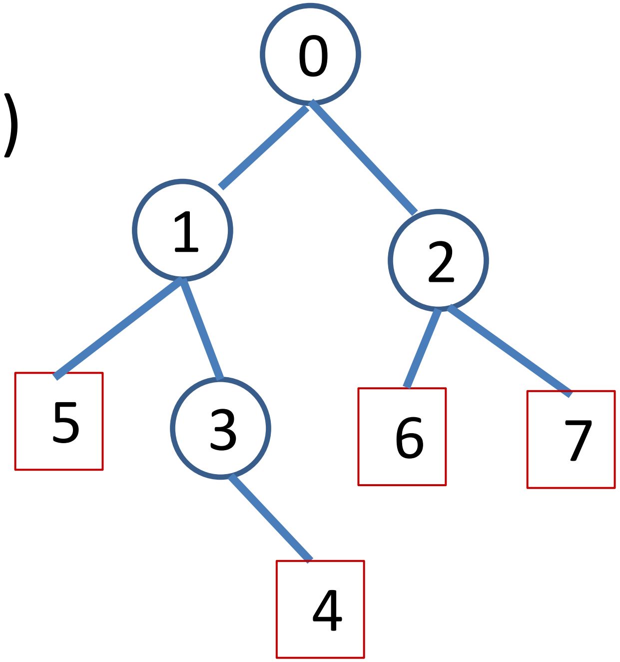
Trees

- Trees are a natural data structure for representing specific data.
  - Family trees.
  - Organizational chart of a corporation, showing who supervises who.
  - Folder (directory) structure on a hard drive.

Terminology

- Root: 0
- Path: 0-2-6, 1-3, 1-3-4
- Parent vs child
- Ascendants vs descendants:
  - ascendants of 3: 1,0 // descendants of 1: 5, 3,4
- Internal vs external nodes

  (non-terminal vs terminal nodes)
- Leaves: 5,4,6,7
- M-ary trees (Binary trees)
- General trees
- Subtree

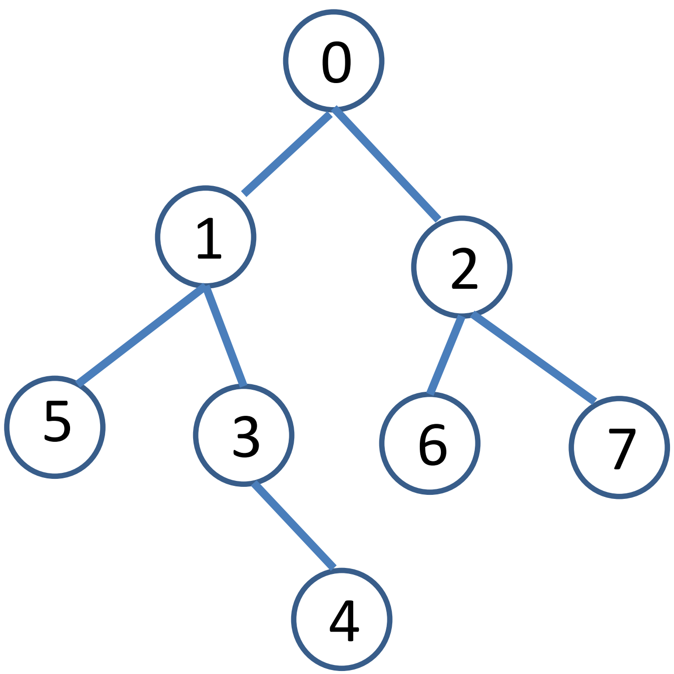


Terminology

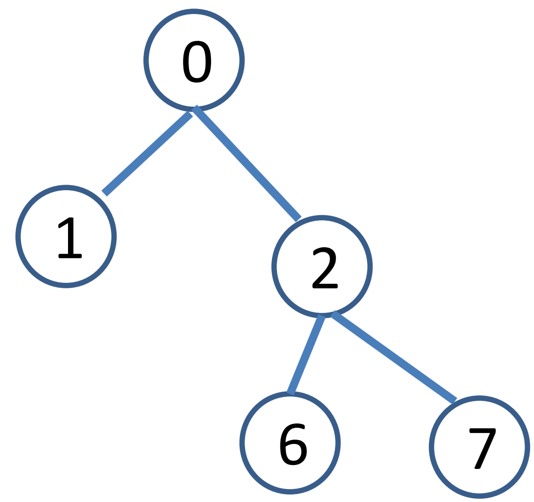
- If Y is the parent of X, then X is called a **child** of Y.
  - The root has no parents.
  - Every other node, except for the root, has exactly one parent.
- A node can have 0, 1, or more children.
- Nodes that have children are called **internal nodes** or **non-terminal nodes**.
- Nodes that have no children are called **terminal nodes**, **external nodes**, or **leaves**.

M-ary Trees - Worksheet

- An **M-ary tree** is a tree where every node is either a leaf or it has **exactly** M children.
- Example: **binary** trees, **ternary** trees, ...



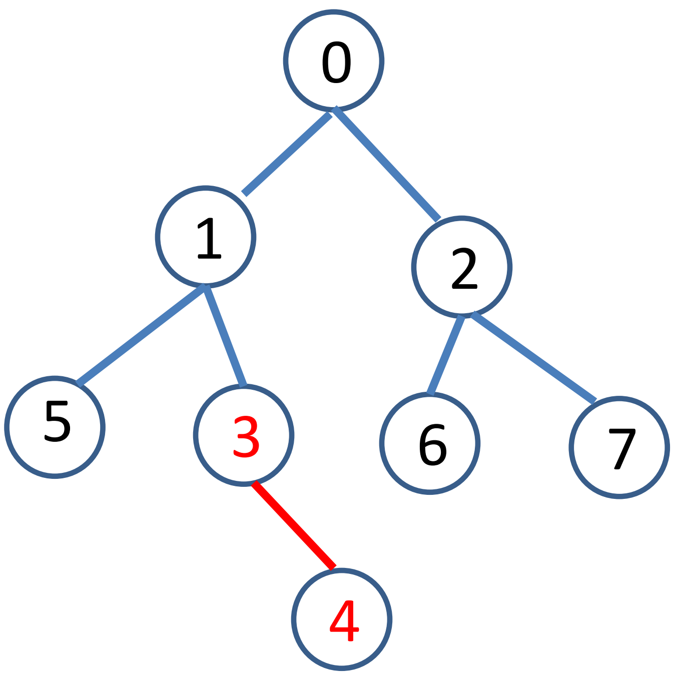
Is this a binary tree?



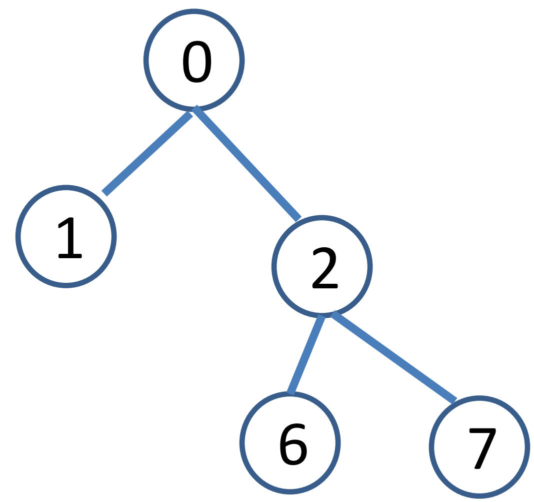
Is this a binary tree?

M-ary Trees - Answers

- An **M-ary tree** is a tree where every node is either a leaf or it has **exactly** M children.
- Example: **binary** trees, **ternary** trees, ...



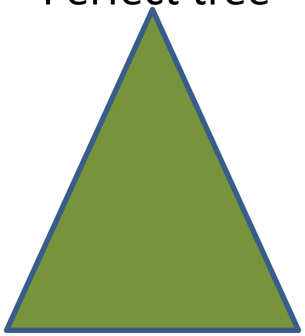
This is **not** a binary tree, node 3 has 1 child.



This is a binary tree.

Types of binary trees

Perfect tree



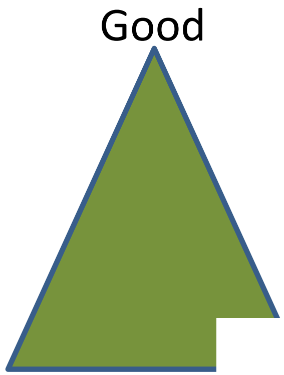
- **Perfect** – each internal node has exactly 2 children and all the leaves are on the same level.
  - E.g. ancestry tree (anyone will have exactly 2 parents).
- **Full** – every node has exactly 0 or 2 children.
  - E.g. tree generated by the Fibonacci recursive calls.
  - *Binary* tree.
- **Complete tree** – every level, except for possibly the last one is completely filled and on the last level, all the nodes are as far on the left as possible.
  - E.g. the heap tree.
  - Height: $\lfloor \lg N \rfloor$ and it can be stored as an array.

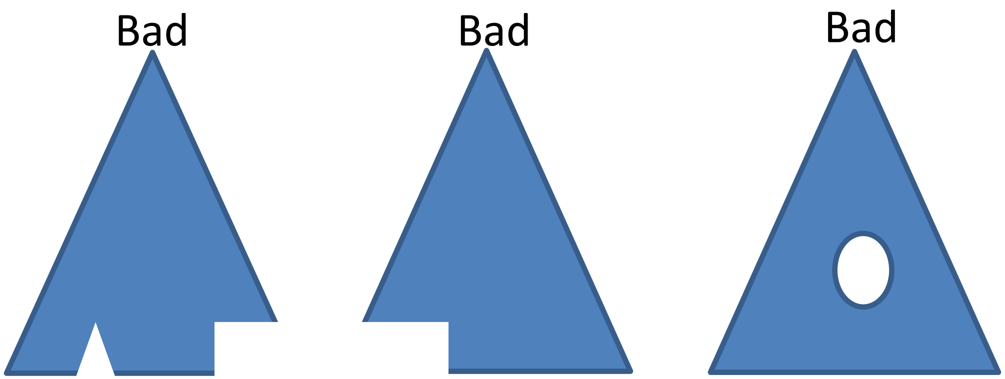
Complete tree

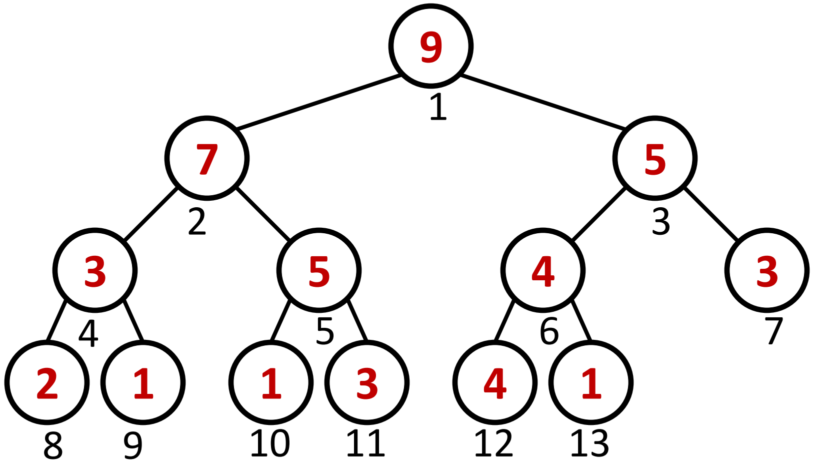
Reference: wikipedia (https://en.wikipedia.org/wiki/Binary_tree)

Complete Tree

– All levels are full, except possibly for the last level.
– At the last level:
  - Nodes are on the left.
  - Empty positions are on the right.
– There is "no hole"







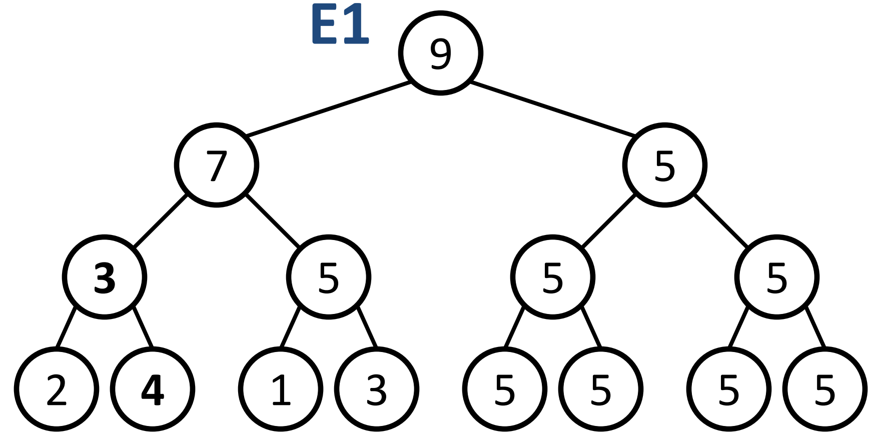
Worksheet

- Self study: Give examples of trees that are:
  - Perfect
  - Full but not complete
  - Complete but not full
  - Neither full not complete

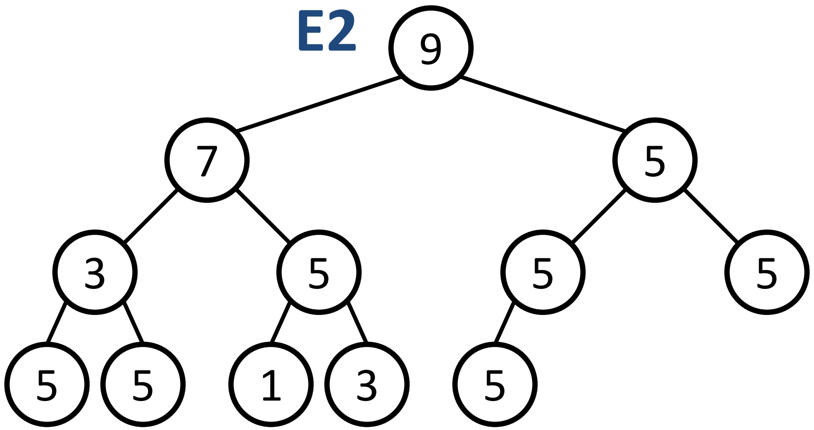
Worksheet

For each tree, say if it is full, complete or perfect.

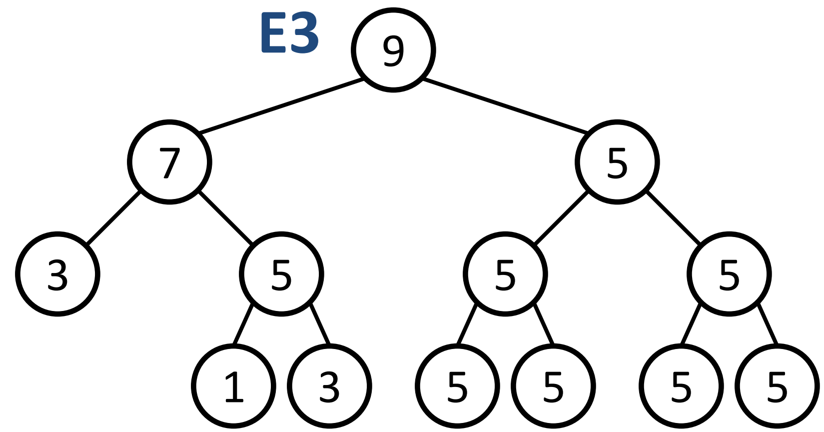
E1



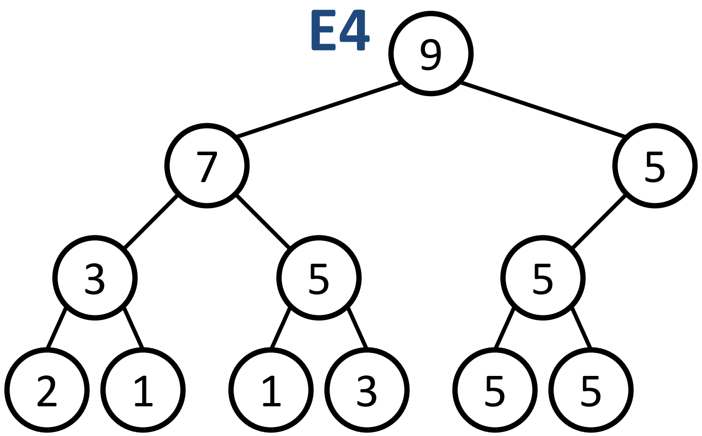
E2



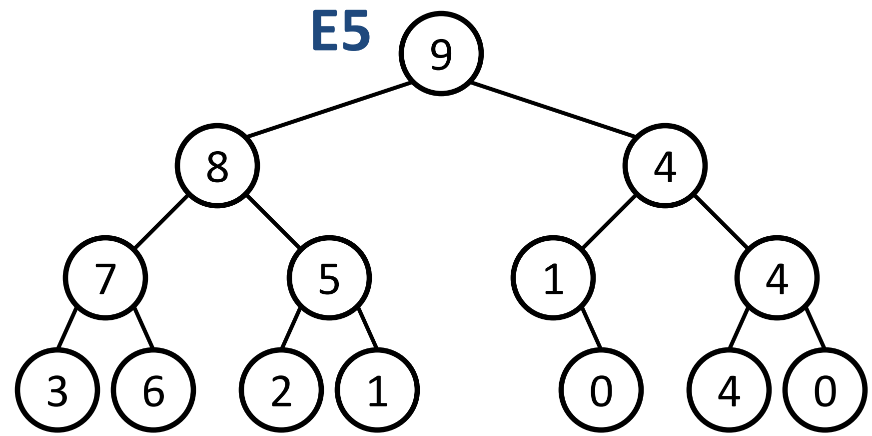
E3



E4



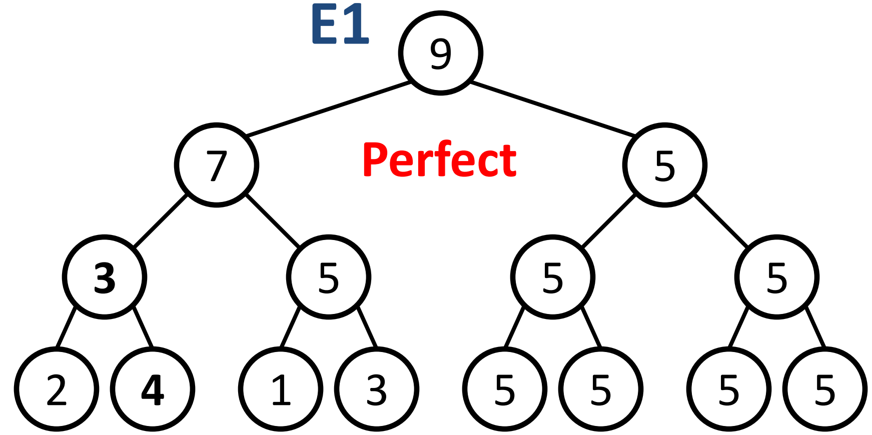
E5



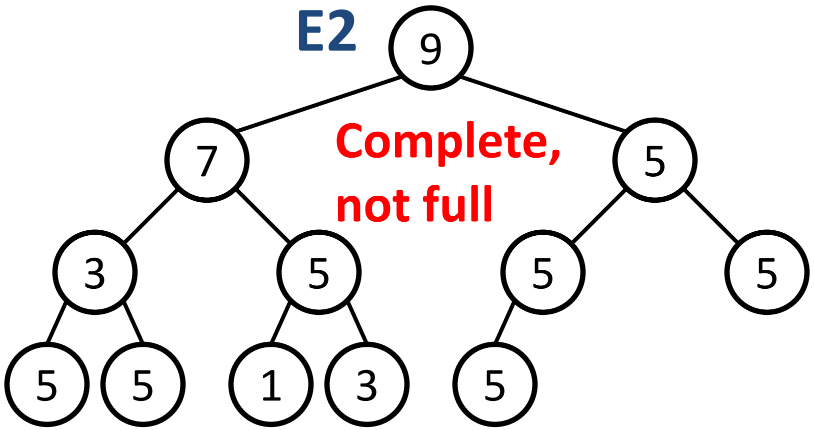
Answers

For each tree, say if it is perfect, full or complete.

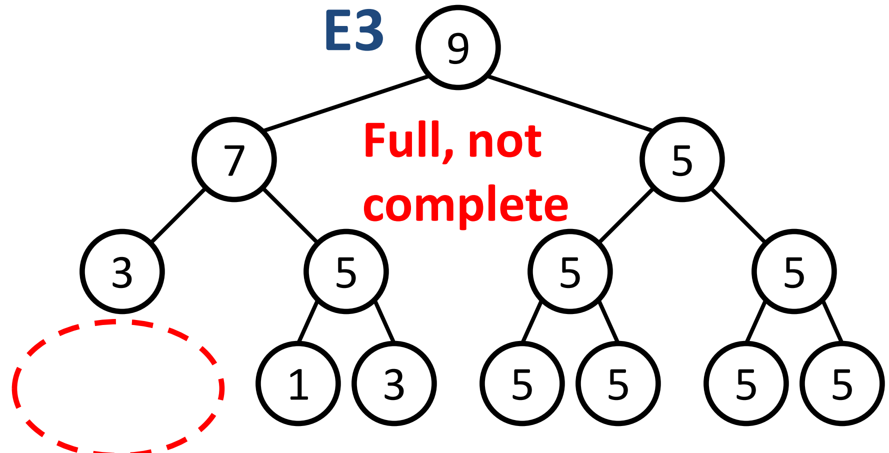
E1



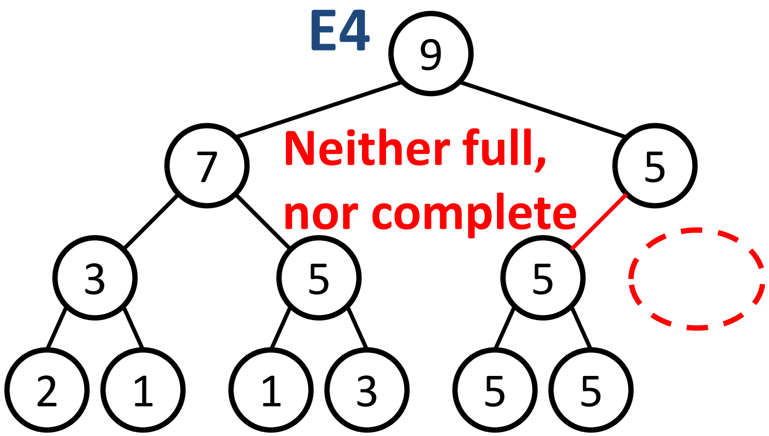
E2



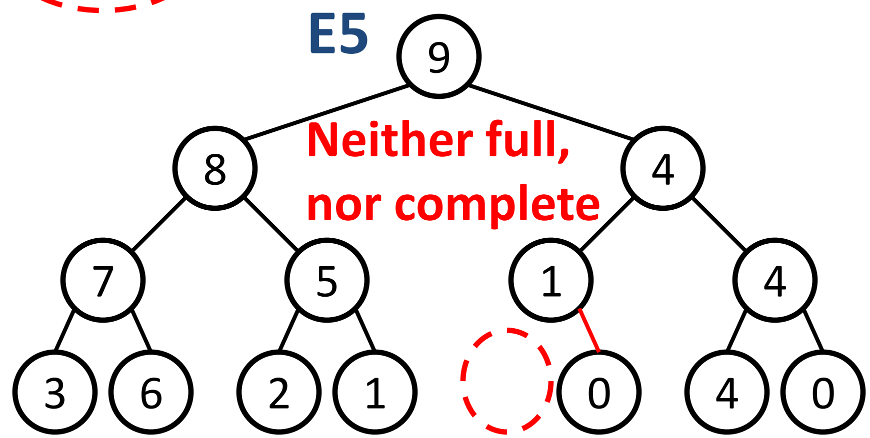
E3



E4

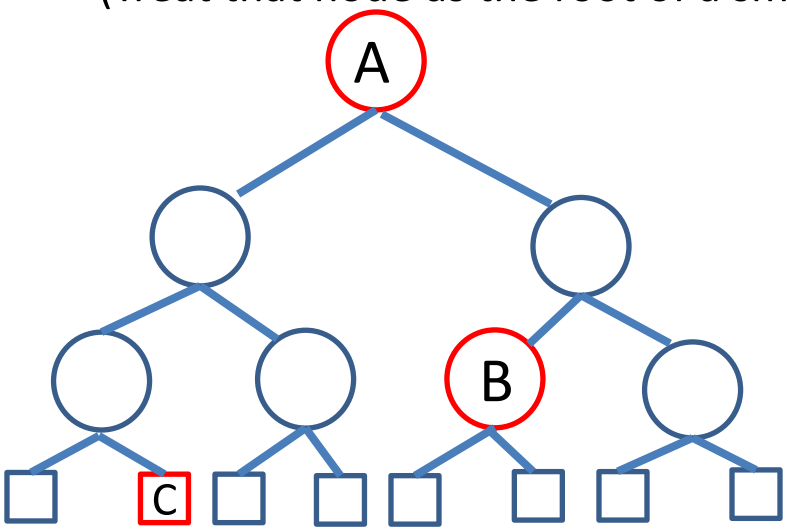


E5



Terminology - Worksheet

- The ***level*** of the root is defined to be 0.
- The ***level*** of each node is defined to be 1+ the level of its parent.
- The ***depth*** of a node is the number of edges from the root to the node.
  (It is equal to the level of that node.)
- The ***height*** of a node is the number of edges *from the node to the deepest leaf*.
  (Treat that node as the root of a small tree)

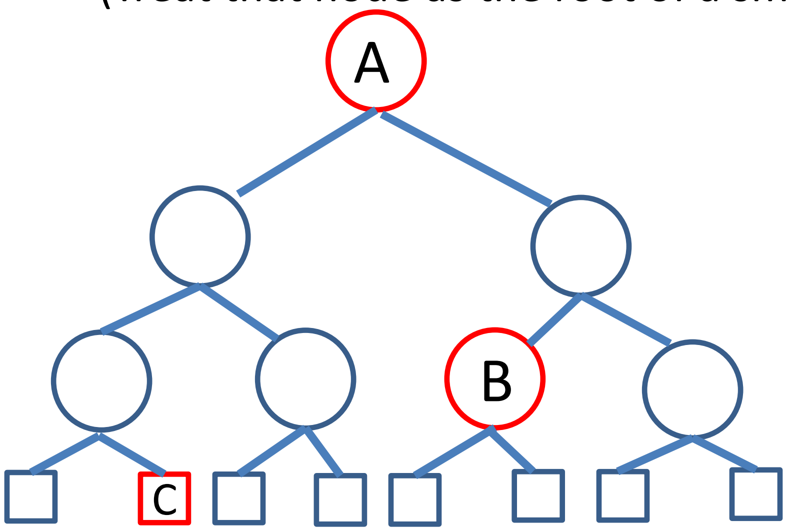


Node	level	depth	height
A			
B			
C			

Practice:
- Give the level, depth and height for each of the red nodes.
- What kinds of tree is this (perfect/full/complete)? ________________________
- How many nodes are on each level? _____________________________________

Terminology - Answers

- The ***level*** of the root is defined to be 0.
- The ***level*** of each node is defined to be 1+ the level of its parent.
- The ***depth*** of a node is the number of edges from the root to the node.
  (It is equal to the level of that node.)
- The ***height*** of a node is the number of edges *from the node to the deepest leaf*.
  (Treat that node as the root of a small tree)



Node	level	depth	height
A	0	0	3
B	2	2	1
C	3	3	0

For any node,
depth + height = tree height

Practice:
- Give the level, depth and height for each of the red nodes.
- What kinds of tree is this (perfect/full/complete)? ____perfect____________
- How many nodes are on each level? _________1, 2, 4, 8___________________

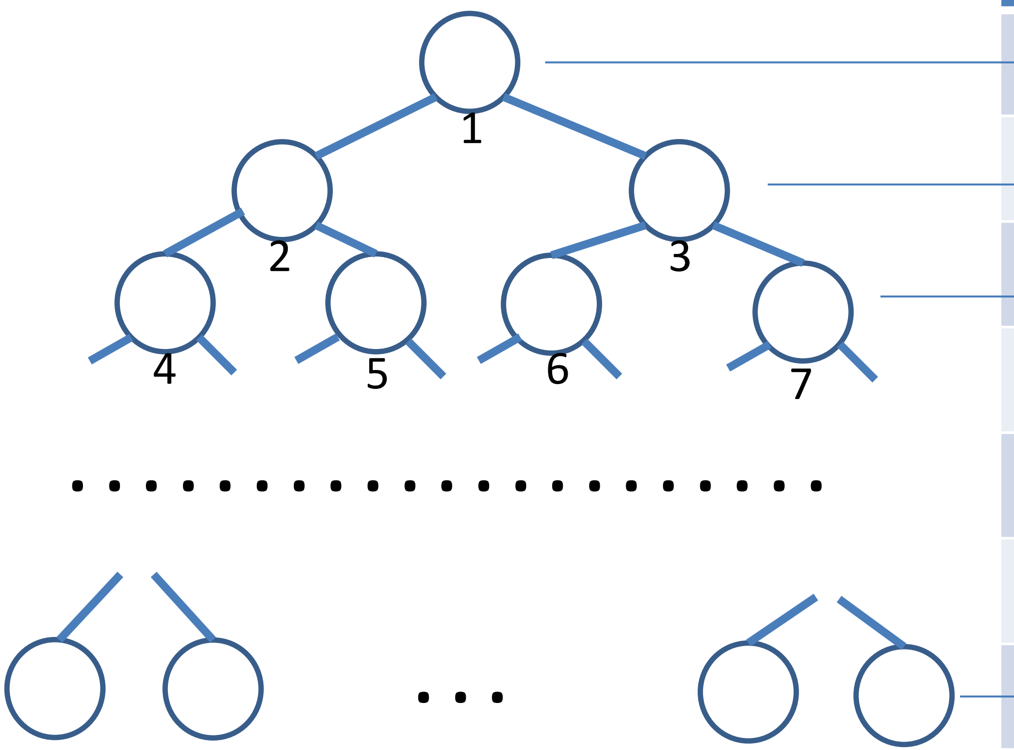
Perfect Binary Trees

A **perfect binary tree** with N nodes and height h, has:

- $\lceil N/2 \rceil$ leaves (half the nodes are on the last level)
- $\lfloor N/2 \rfloor$ internal nodes (half the nodes are internal)
- Height : $h = \lfloor \lg N \rfloor$
- Levels : $\lfloor \lg N \rfloor + 1$ (= lg(N+1))

$$\sum_{k=0}^{h} 2^k = 2^{h+1} - 1$$

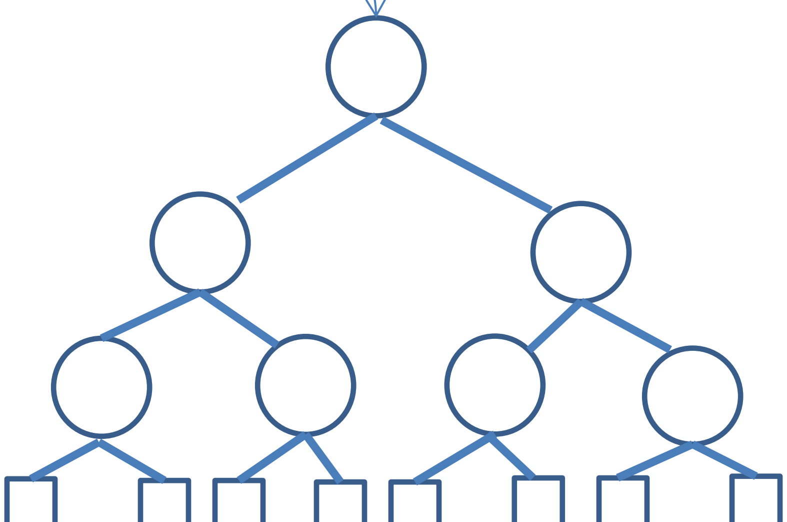
In the other direction: $N = 2^{h+1} - 1$

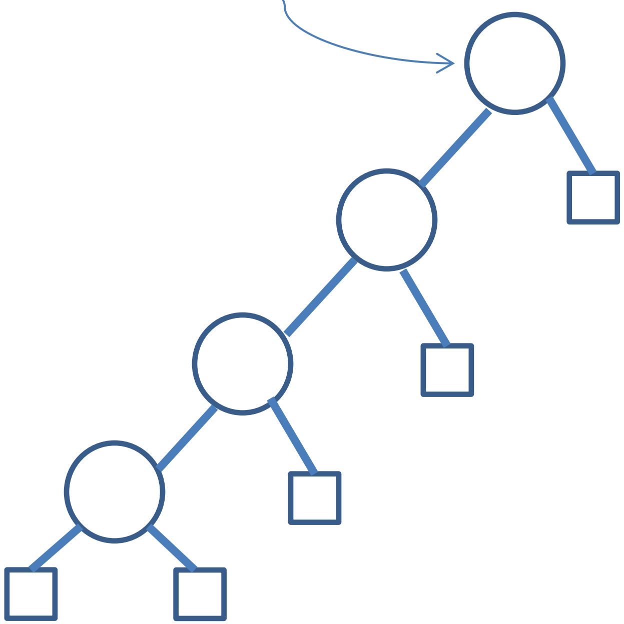


Level	Nodes per level	Sum of nodes from root up to this level
0	2^0 (=1)	$2^1 - 1$ (=1)
1	2^1 (=2)	$2^2 - 1$ (=3)
2	2^2 (=4)	$2^3 - 1$ (=7)
…	…	
i	2^i	$2^{i+1} - 1$
…	…	
h	2^h	$2^{h+1} - 1$

Properties of Full Trees

- A **full** binary tree (0/2 children) with P internal nodes has:
  - P+1 external nodes.
  - 2P edges (links).
  - height at least lg P and at most P:
    - lg P if all external nodes are at the same level (perfect tree)
    - P if each internal node has one external child.





Proof

Prove that a full tree with P internal nodes has P+1 external leaves.

- Full tree property:
- What proof style will you use?