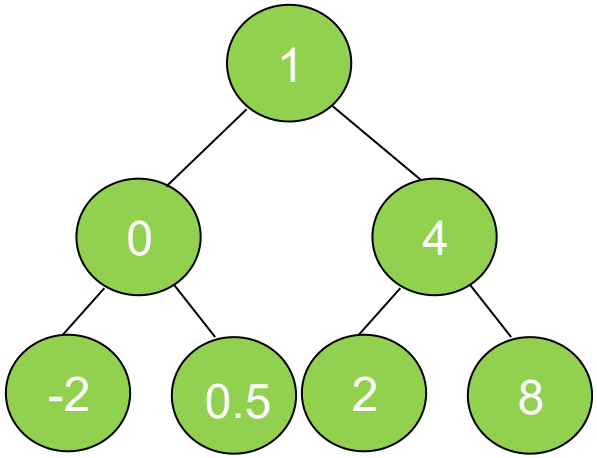
Lecture 8
Binary Search Tree
Exercises ANS

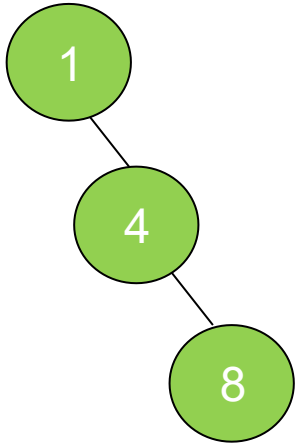
Department of Computer Science
Hofstra University

Binary Tree

- For a binary tree of height h, what is its minimum and maximum number of leaves and total nodes?
- When it is a full binary tree, it has
  - Max number of leaves: 2^h
  - Max number of nodes: $2^{(h+1)} - 1$
- When it is a linked list:
  - Min number of leaves: 1
  - Min number of nodes: h + 1



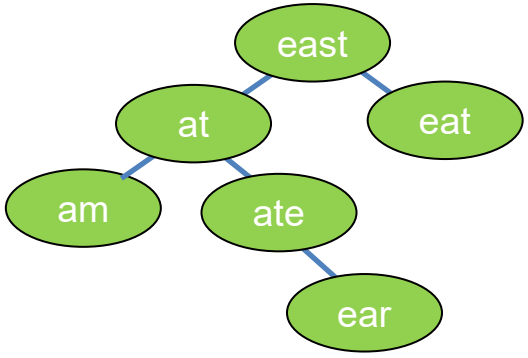
h=2



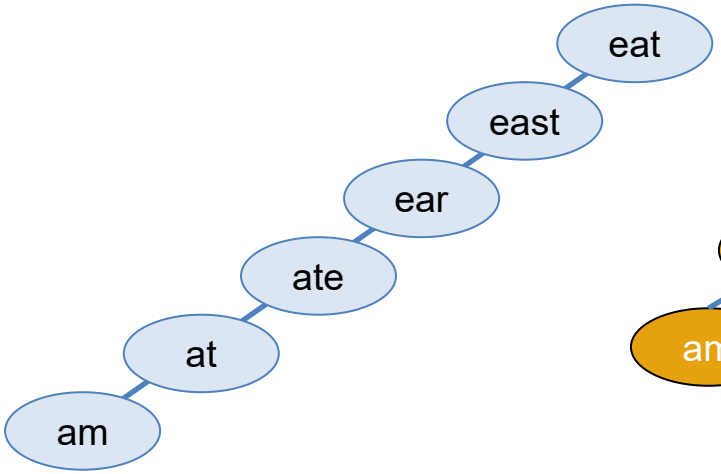
h=2

AVL Tree

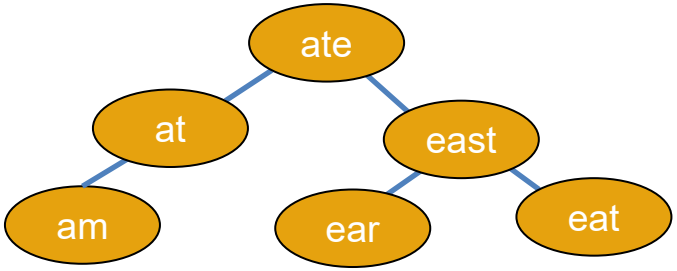
- Which is an AVL tree (Balanced BST)?
- ANS: c. Only it satisfies the AVL invariant for all nodes:
- **| LeftHeight – RightHeight | <=1**



(a)



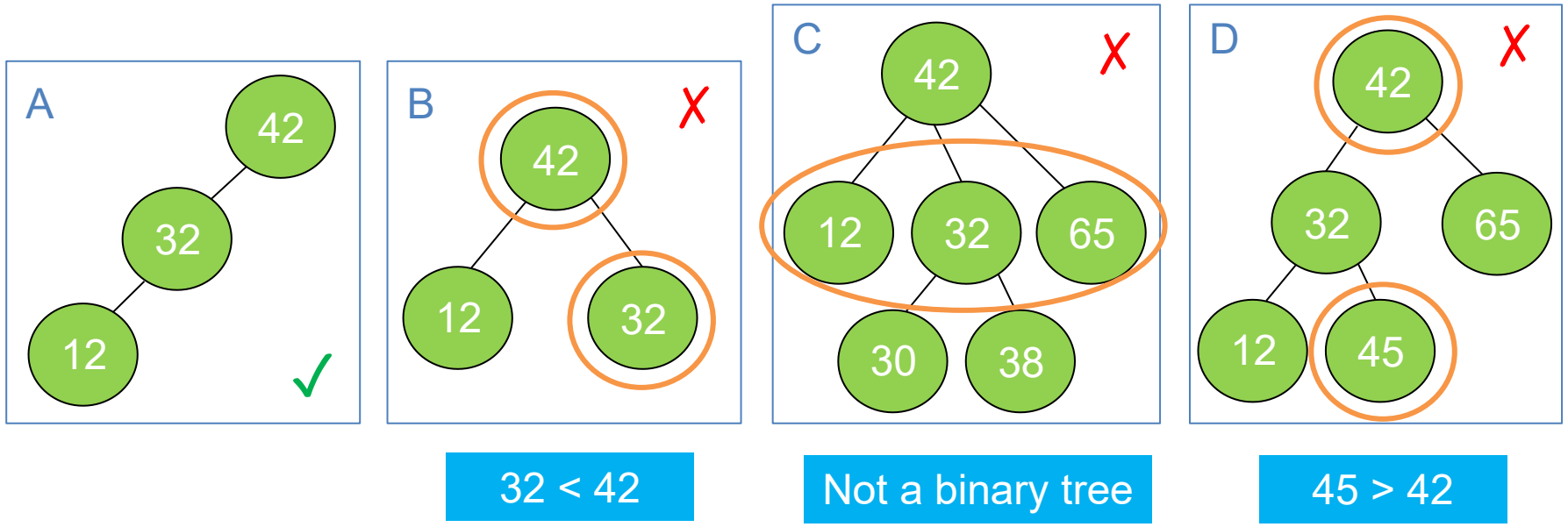
(b)



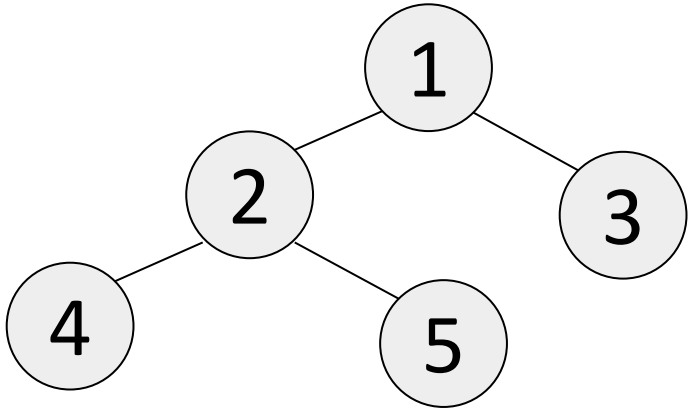
(c)

Binary Search Tree (BST)

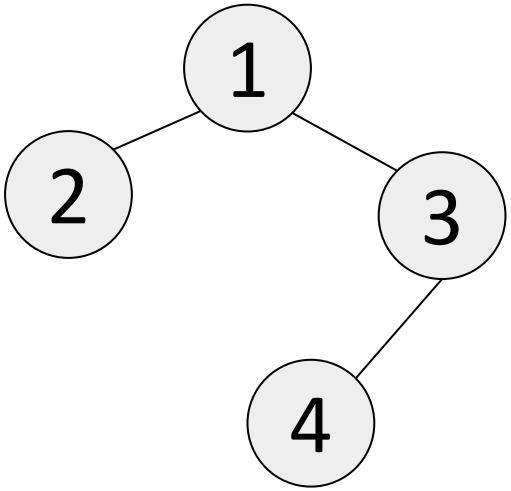
- Which of the following is a BST?
- ANS: A



Pre, In and Post Order Traversal

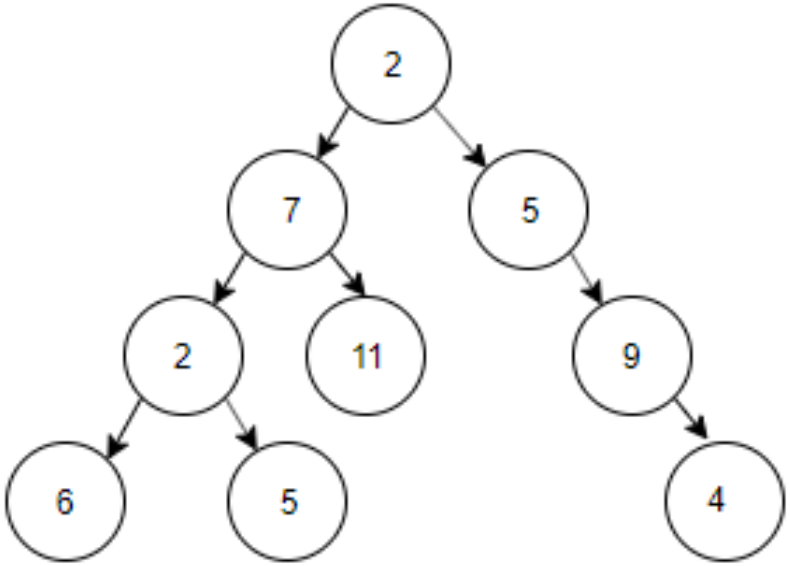


- Pre-Order: 12453
- In-Order: 42513
- Post-Order: 45231



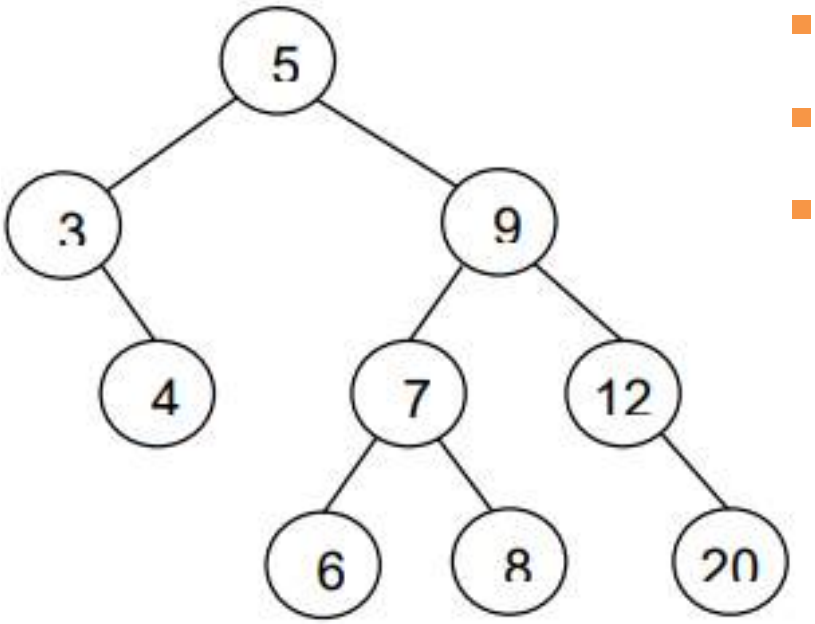
- Pre-Order: 1234
- In-Order: 2143
- Post-Order: 2431

Pre, In and Post Order Traversal



- Pre-Order: 2, 7, 2, 6, 5, 11, 5, 9, 4
- In-Order: 6, 2, 5, 7, 11, 2, 5, 9, 4
- Post-Order: 6, 5, 2, 11, 7, 4, 9, 5, 2

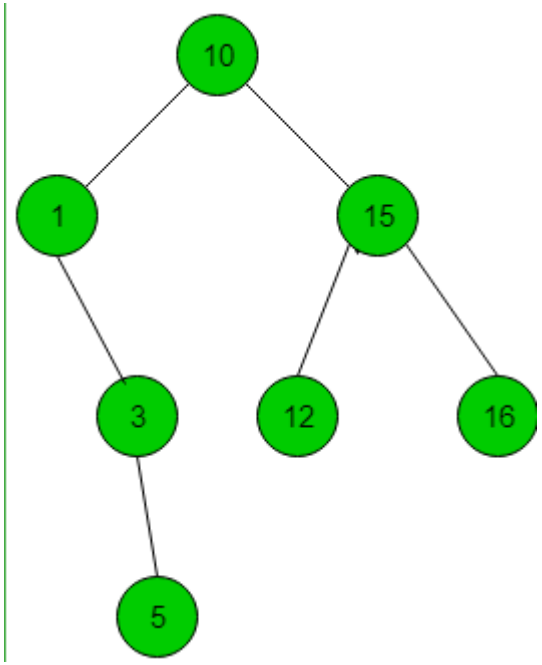
Pre, In and Post Order Traversal



- Preorder [5, 3, 4, 9, 7, 6, 8, 12, 20]
- In-order: [3, 4, 5, 6, 7, 8, 9, 12, 20]
- Post-order: [4, 3, 6, 8, 7, 20, 12, 9, 5]

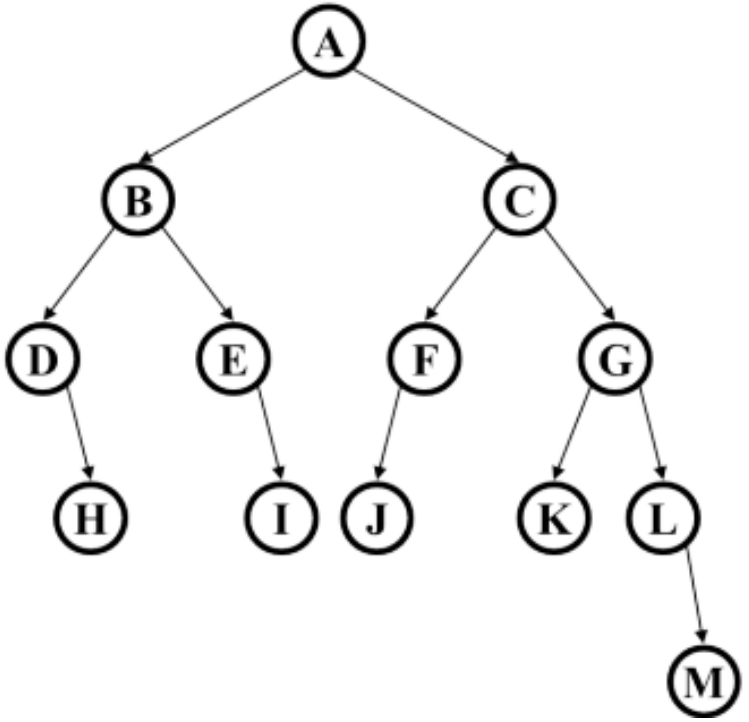
BST

- The following numbers are inserted into an empty binary search tree in the given order: 10, 1, 3, 5, 15, 12, 16. What is the height of the binary search tree (the height is the maximum distance of a leaf node from the root, i.e. a tree with a single root node has height 0.)?
- ANS: 3



BST

- Assume this tree is a binary search tree. What is the maximum number of nodes that could be added to the tree without increasing its height?
- ANS: 18
- A full binary tree with height h has a total number of nodes given by the formula: $n = 2^{h+1}-1 = 2^{4+1}-1=31$, since the height is 4.
- Currently there are 13 nodes, so you can add 31-13=18 nodes without increasing its height.
- You can also count the inserted nodes until the tree is full.



BST

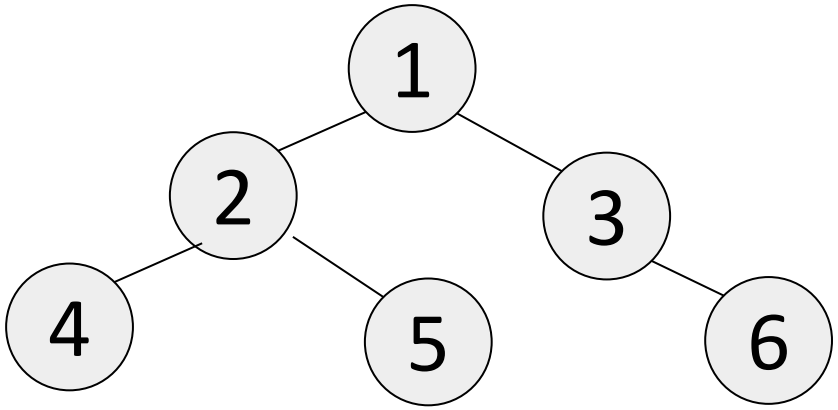
- Suppose the numbers 7, 5, 1, 8, 3, 6, 0, 9, 4, 2 are inserted in that order into an initially empty binary search tree. What is the in-order traversal sequence of the resultant tree?
- ANS: 0 1 2 3 4 5 6 7 8 9
- In-order traversal of a binary search tree visits the nodes in ascending order of their values. You do not need to construct the tree.

Quiz: Tree Derivation

- Given: Pre-order traversal of nodes is 1, 2, 4, 5, 3, 6; In-order traversal of nodes is 4, 2, 5, 1, 3, 6. What is the post-order traversal of nodes?

Quiz: Tree Derivation ANS

- For a binary tree, its pre-order traversal of nodes is 1, 2, 4, 5, 3, 6; its in-order traversal of nodes is 4, 2, 5, 1, 3, 6. Construct the tree. What is the post-order traversal of nodes?
- ANS: we know 1 is the tree root from pre-order traversal. In-order traversal is 4, 2, 5, 1, 3, 6, so we know left subtree has nodes 4,2,5, and right subtree has nodes 3,6.
- For the left subtree, pre-order traversal is 2, 4, 5, so we know 2 is the subtree root; In-order traversal is 4, 2, 5, so we know left subtree has node 4, and right subtree has node 5.
- For the right subtree, pre-order traversal is 3, 6, so we know 3 is the subtree root; In-order traversal is 3, 6, so we know left subtree has no node, and right subtree has node 6.
- We can draw the tree now and derive the post order traversal 4, 5, 2, 6, 3, 1



- **Pre-order traversal**:
  - Begins at the root, ends at the right-most node
- **In-order traversal**:
  - Begins at the left-most node, ends at the rightmost node
- **Post-order traversal**:
  - Begins with the left-most node, ends with the root

Quiz: Tree Derivation II

- For a binary tree, its pre-order traversal of nodes is ABCDEFG; its in-order traversal of nodes is CDBAEGF. Construct the tree. What is the post-order traversal of nodes?

Quiz: Tree Derivation II ANS

- For a binary tree, its pre-order traversal of nodes is ABCDEFG; its in-order traversal of nodes is CDBAEGF. Construct the tree. What is the post-order traversal of nodes?
- 1. Root Identification: The first element in pre-order (A) is the root.
- 2. Subtree Division:
- - In in-order traversal, elements before A (CDB) form the left subtree, and elements after (EGF) form the right subtree.
- 3. Left Subtree Construction:
- - Pre-order segment for left: BCD
- - Root is B (first in pre-order). In in-order (CDB), left of B is CD and right is empty.
- - B's left child: C (next in pre-order). In in-order (CD), C has a right child D.
- 4. Right Subtree Construction:
- - Pre-order segment for right: EFG
- - Root is E (first in pre-order). In in-order (EGF), E has a right segment GF.
- - E's right child: F (next in pre-order). In in-order (GF), F has a left child G.
- We can draw the tree now and derive the post order traversal DCBGFEA.