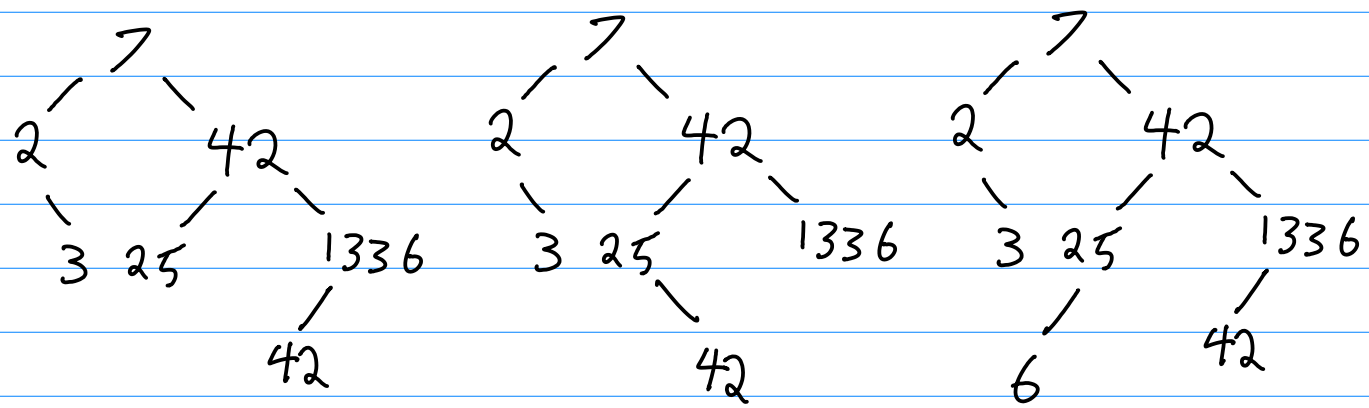
CPSC 220 9/16

<u>Binary Search Tree</u>: binary tree where for every node the left sub-tree only contains nodes with a value less than the node and the right subtree only contains nodes with a value greater than or equal to the node



Is BST ✓ ✗ ✗

BST Operations

Add: if tree is empty add new node as root
otherwise make root current node
* if new node is less than current node
if left child exists
make left child current node, go to *
if left child does not exist
add new node as left child of current node
if new node is greater than or equal to current node
if right child exists
make right child current node, go to *
if right child does not exist
add new node as right child of current node

Find: make root current node
* if desired node == current node
return current node
if desired node is less than current node
if left child exists
make left child current node, go to *
if left child does not exist
return node not found
if desired node is greater than or equal to current node
if right child exists
make right child current node, go to *
if right child does not exist
return node not found