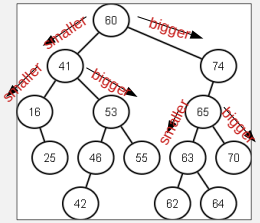
# Binary Search Trees

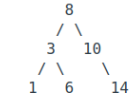
- a binary tree with an ordering property for the elements
  - for every node –
    - all of the elements in the left subtree are less than or equal to the node's element
    - all of the elements in the right subtree are greater than the node's element
- operations
  - lookup
  - insert
  - remove
  - visit all elements (traverse) in order

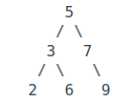



(dummy leaves not shown)

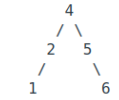


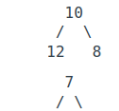
Which of the following are valid binary search trees? Choose all that apply.

- [x]


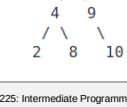

- [ ]



6 is bigger than 5 but in its left subtree

- [x]



- [ ]



12 is bigger than 10 but in its left subtree
8 is smaller than 10 but in its right subtree

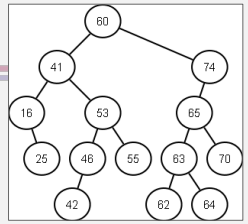
- [ ]



8 is bigger than 7 but in its left subtree



# Binary Search Trees

- lookup
  - moving down, 1-finger (only go to one child) pattern → loop
  - search ends when element is found or a leaf is reached (element not found)




(dummy leaves not shown)

- insert
  - can only insert at a leaf
  - the correct insertion point is the leaf where an unsuccessful search for the element ends up
- remove
  - can only remove above a leaf
  - if the element to remove does not have at least one leaf child, swap it with a safe element which does has at least one leaf child
    - i.e. the next element larger or smaller than the one to remove