

# CS 477/677 Analysis of Algorithms

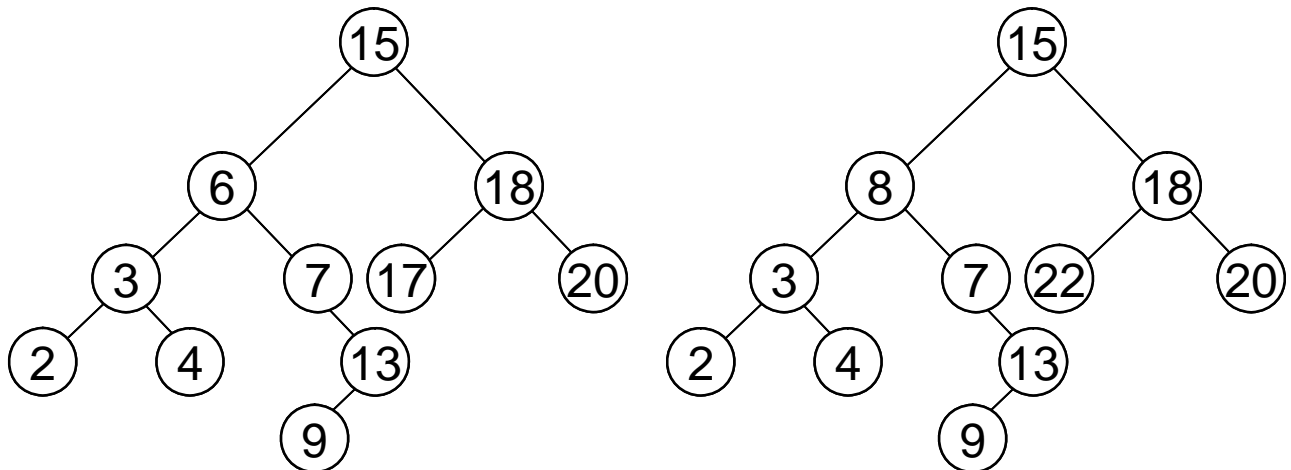
## Homework 5

Due October 21

**Note 1:** Students in the CS 477 section must solve problems 1 through 3 and can solve either problem 4 or 5 for extra credit. Students in the 677 section must solve problems 1 through 4 and can solve problem 5 for extra credit.

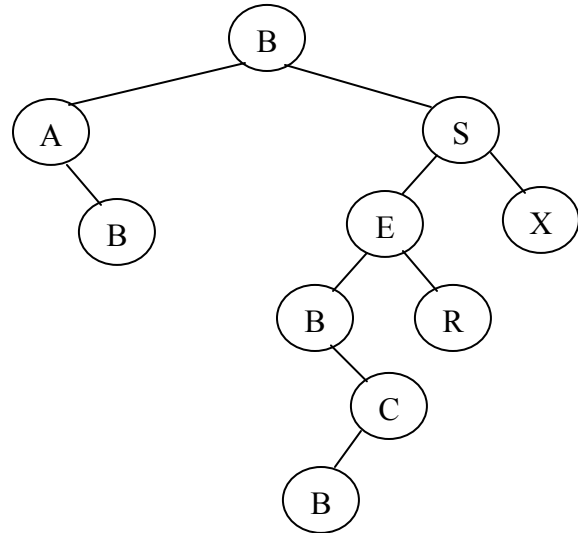
**Note 2:** For the programming problems below, include in your hardcopy submission a listing of your algorithm and of the output. Also send your source code as an attachment to [rkelly@cse.unr.edu](mailto:rkelly@cse.unr.edu). The subject line of the email should be HW5.

**1. [40 points]** Implement in C/C++ a linear time (i.e.,  $\Theta(n)$ ) algorithm that determines whether a binary tree is a binary search tree. Your algorithm should return TRUE if the tree is a binary search tree, otherwise it should return FALSE and also display the value



of a node where the binary search tree property is not met. Print and submit the output of your algorithm on the following trees (you can create the trees “by hand”, by adding left and right pointers to children – no need to use TreeInsert operations).

2. [40 points] Implement in C/C++ an algorithm that counts and returns the number of items in a binary search tree that are equal to a given key. Print and submit the results of your algorithm for the tree shown at right and for the key equal to B.



3. [20 points] What is the difference between the binary-search-tree property and the heap property? Can the heap property be used to print out the keys of an n-node tree in sorted order in  $O(n)$  time? Explain how or why not.

4. [Graduate only – 20 points]

Exercise 12.2-5 (page 260).

Exercise 12.2-1 (page 259).

5. [Extra credit – 20 points]

Exercise 6.1-1 (page 129).

Exercise 6.1-2 (page 129). Note: this problem could be solved using the result from Exercise 6.1-1 above.