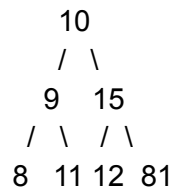


1. What is the difference between a binary tree and a binary search tree?
2. True or false: All binary trees are binary search trees, but not all binary search trees are binary trees.
3. What is the average case of insertion and searching in binary search trees?
4. What is the worst case? When does this worst case occur?
5. What is a leaf? A node? A parent? A child? A root node? An interior node?
6. What is the order of infix traversal?
7. Draw the binary search tree that would occur from inserting the values [10, 5, 15, 1, 19, 13, 9]
8. Perform an infix tree traversal on the tree from question 7.
9. Is the following tree a binary search tree?



10. When do you use comparable and when do you use a comparator?
11. True or False: A comparator is a separate class that implements an order other than natural ordering?
12. What method do you need to implement if a class is comparable?
13. What method do you need to implement if a class is a comparator?
14. If a = "Hello" and b = "Goodbye" what will a.compareTo(b) return?
15. If a = "Hello" and b = "Hello" what will a.compareTo(b) return?
16. If a = "Hello" and b = "Hellos" what will a.compareTo(b) return?
17. What does it mean for a tree to be bushy? What about branchy?