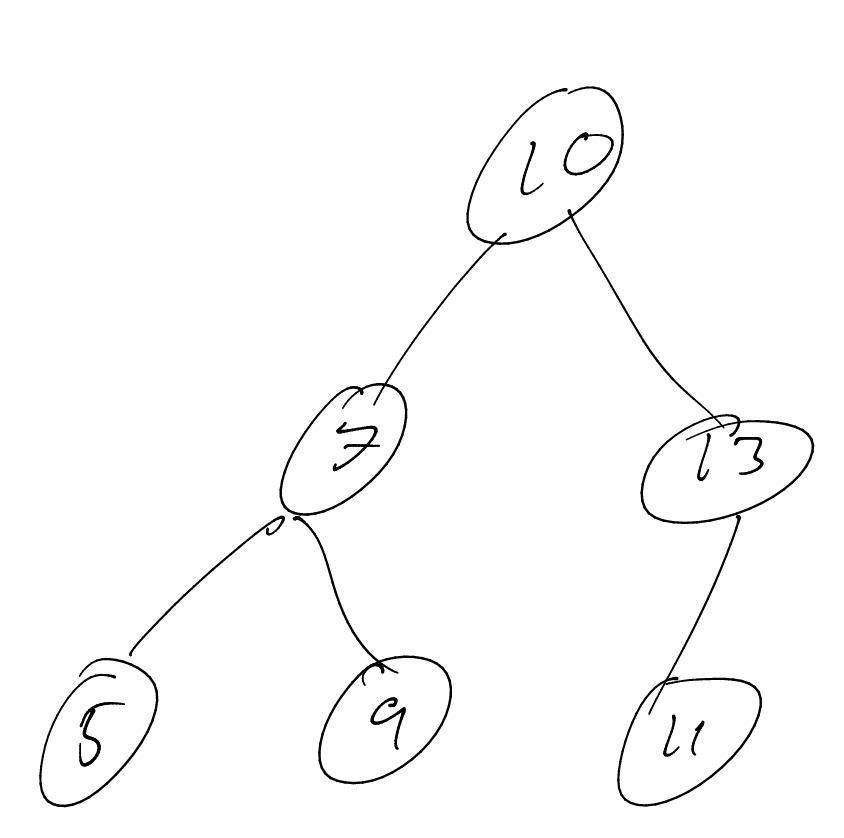
**Binary Tree Activity**

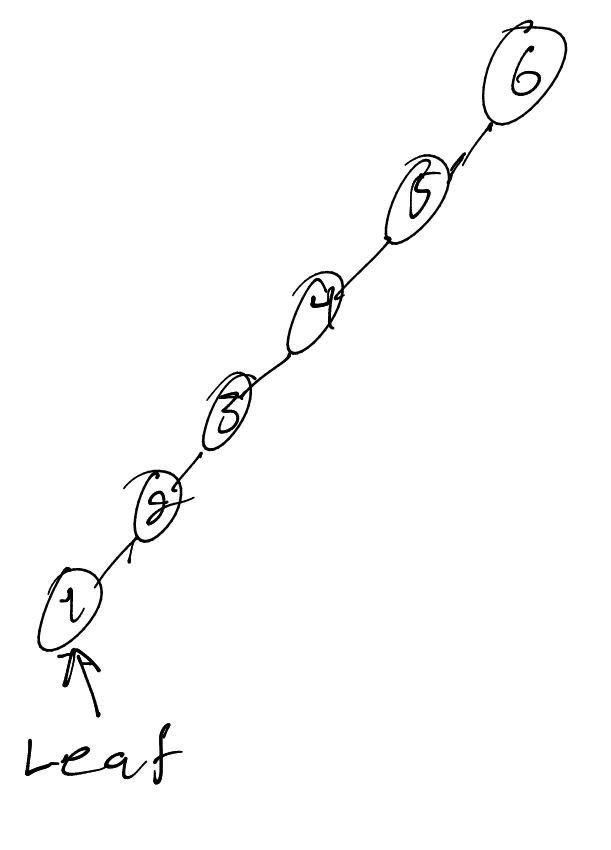
1. Draw a binary tree with 6 Nodes where there are exactly 3 leaves.



1. What is the height of the above tree?

**3**

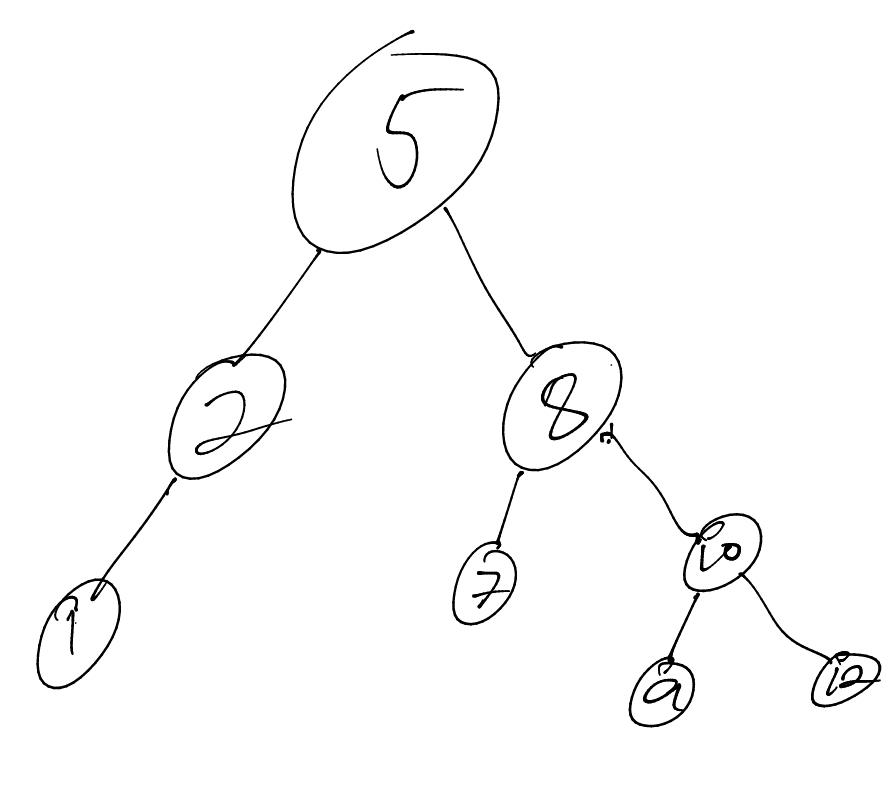
1. Draw a binary tree with 6 Nodes where there is exactly 1 leaf.



1. What is the height of the above tree?

6

1. Insert the following values into an empty tree. Draw what the tree would look like. Values: {5, 8, 2, 1, 10, 7, 9, 12 }



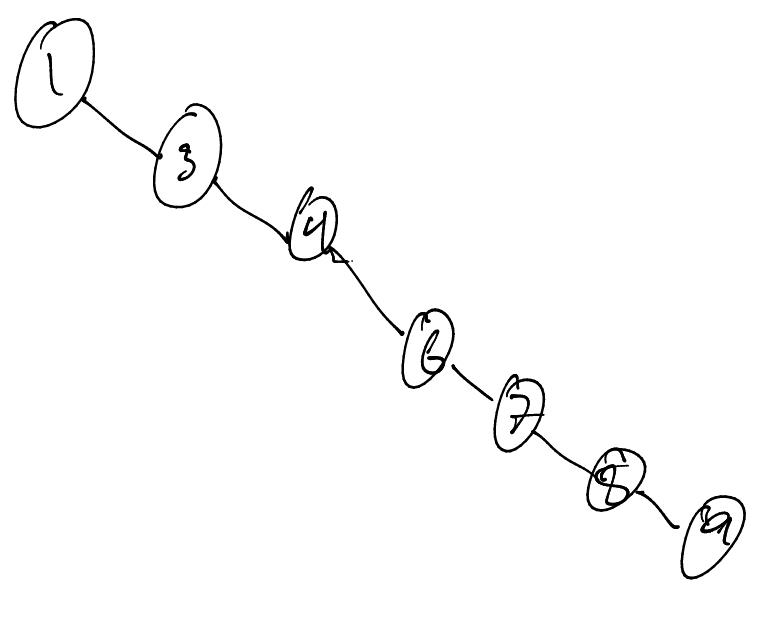
1. Which nodes would be examined in order to search for the value 7?
   1. 5 and 8
2. Which nodes would be examined in order to search for the value 3?
   1. 5 and 2
3. What would be the Time Complexity (Big O notation) of this search?
   1. O(2) Linear (O(n))
4. What is a Pre Order Traversal of this tree?

5, 2, 1, 8, 7, 10, 9, 12

1. What is a Post Order Traversal of this tree?

1, 2, 7, 9, 12, 10, 8, 5

1. Draw the following tree with the following values inserted: Values: {1, 3, 4, 6, 7, 8, 9}



1. Which nodes would be examined in order to search for the value 7?

1, 3, 4, and 6

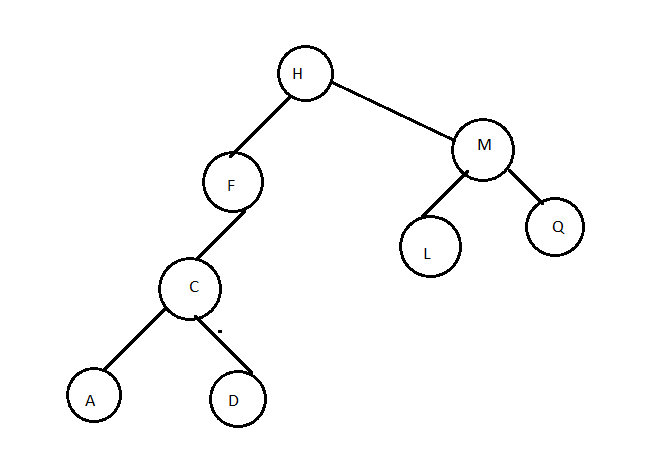
1. What would be the Time Complexity (Big O notation) of this search?

O(4) Linear (still)

1. How does the order the items are added to the tree affect the efficiency of the search?

When all the items are added in order, you have to traverse a lot more nodes, as it has to pass through all the ones either smaller or larger than it. Instead, when you add a new node, or a few new nodes, you could re-order it, finding the median value, and adding them going up and down from that center point.

1. List the pre-order, In-Order, and Post Order traversals of the tree below.
   1. Pre Order: 1, 3, 4, 6, 7, 8, 9
   2. In Order: 1, 3, 4, 6, 7, 8, 9
   3. Post Order: 9, 8, 7, 6, 4, 3, 1



1. Draw what the above tree would look like if the value 'F' were deleted?

