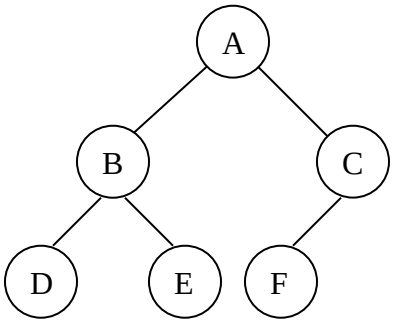
Trees Worksheet

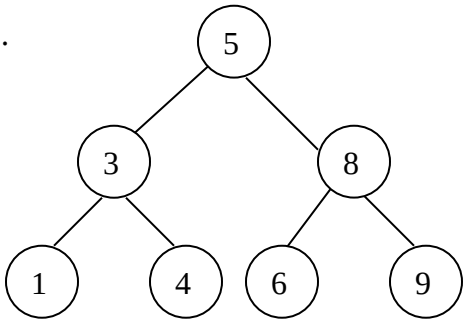
(preparation for `TreeLab`)

1.



number of Nodes	
Height	
Depth of "B"	
Length of longest path	
Root	
number of Leaves	
number of Parents	
number of Only Children	
number of nodes at Level 0	
Preorder Traverse	
Inorder Traverse	
Postorder Traverse	
Levelorder Traverse	

2.



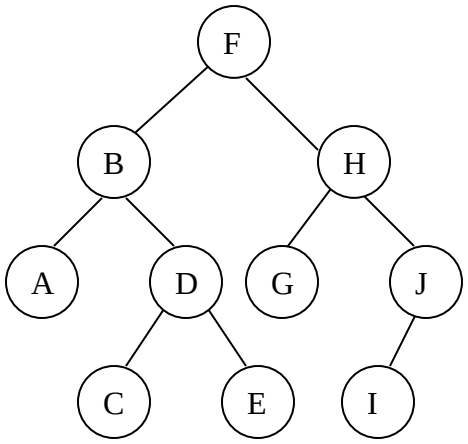
number of Nodes	
Height	
Level of "5"	
Length of longest path	
Root	
number of Leaves	
number of Parents	
number of Only Children	
number of nodes at Level 2	
Preorder Traverse	
Inorder Traverse	
Postorder Traverse	
Levelorder Traverse	

3. Trees are naturally recursive structures. Trace this code to search the BST above for "6".

```
If the tree is empty
   return false
else if the item in the root equals the target
   return true
else if the item in the root is greater than the target
   return the result of searching the root's left subtree
else
   return the result of searching the root's right subtree
```

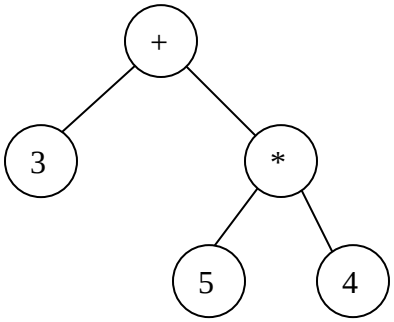
Then trace the code to search the tree above for "7".

4.



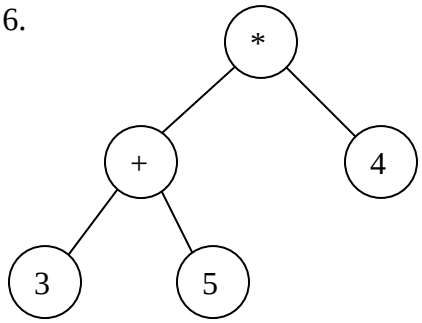
number of Nodes	
Height	
Depth of “D”	
Width	
Root	
number of Leaves	
number of Parents	
number of Only Children	
number of nodes at Level 3	
Preorder Traverse	
Inorder Traverse	
Postorder Traverse	
Levelorder Traverse	

5.



number of Nodes	
Height of Tree	
Depth of “*”	
Width	
Root	
number of Leaves	
number of Parents	
number of Only Children	
number of nodes at Depth 1	
Preorder Traverse	
Inorder Traverse	
Postorder Traverse	
Levelorder Traverse	

6.



number of Nodes	
Height	
Depth of “*”	
Width	
Root	
number of Leaves	
number of Parents	
number of Only Children	
number of nodes at Level 2	
Preorder Traverse	
Inorder Traverse	
Postorder Traverse	
Levelorder Traverse	

7. Which of the trees above are complete? _____

8. Which of the trees above are full? _____

9. Which of the trees above have the Binary Search Tree property? _____

10. Which of the trees above have the heap (either max or min) property? _____

11. Which of the trees above are expression trees? _____
12. Draw some **full binary trees** and fill out the table:

	number of nodes	depth (level) of the last node	height
	1		
	3		
	7		
	15		
	31		
write formulas		d	
	n		

13. A full binary tree with *d* levels can have a maximum of _____ nodes.

14. If a full binary tree has *n* nodes, then it has _____ levels.

15. What is the smallest height needed to store 100,000 nodes in a binary tree?_____

16. Draw an expression tree for the prefix expression `+*123`
    Give its infix expression _____

    and its postfix expression_____

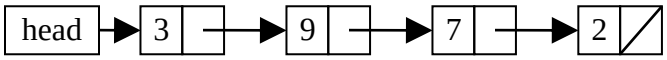
17. Draw an expression tree for the infix expression `2*8+9/3`
    Give its prefix expression _____

    and its postfix expression_____

18. Draw an expression tree for

```
yr % 4 == 0 && (yr % 100 != 0 || yr % 400 == 0)
```

Questions 19-21 use recursion on `ListNode` linked lists. Think recursively!



19.
```
public int countNodes(ListNode head)
  {




  }
```

20.
```
public void display(ListNode head)
  {




  }
```

21.
```
public int findMax(ListNode head)
  {




  }
```

22. Trees are naturally recursive structures. Print the preorder expression on a tree made of `TreeNodes`. Use recursion.

```
public static void preorderTraverse( TreeNode t )
  {
```

23. Open the `TreeLab` shell. Complete the code.