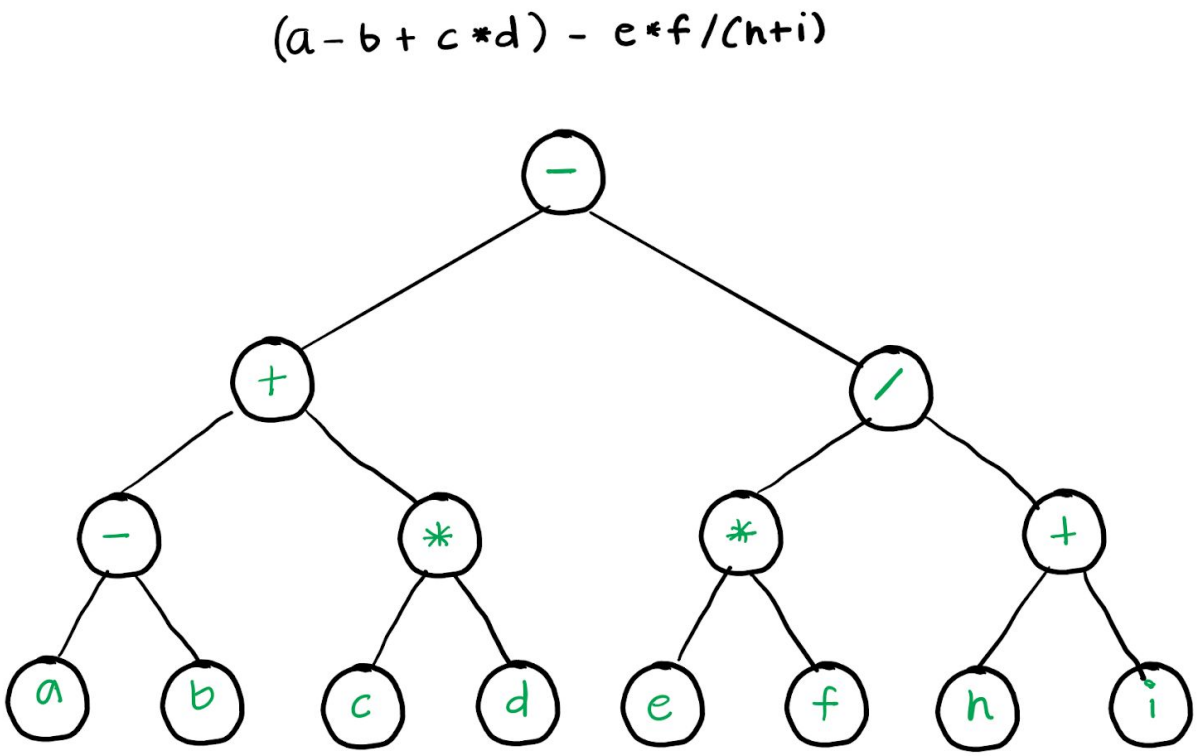
Individual Written Homework Assignment 3: Binary Trees, BSTs, and AVL Trees

I. Problem Set:

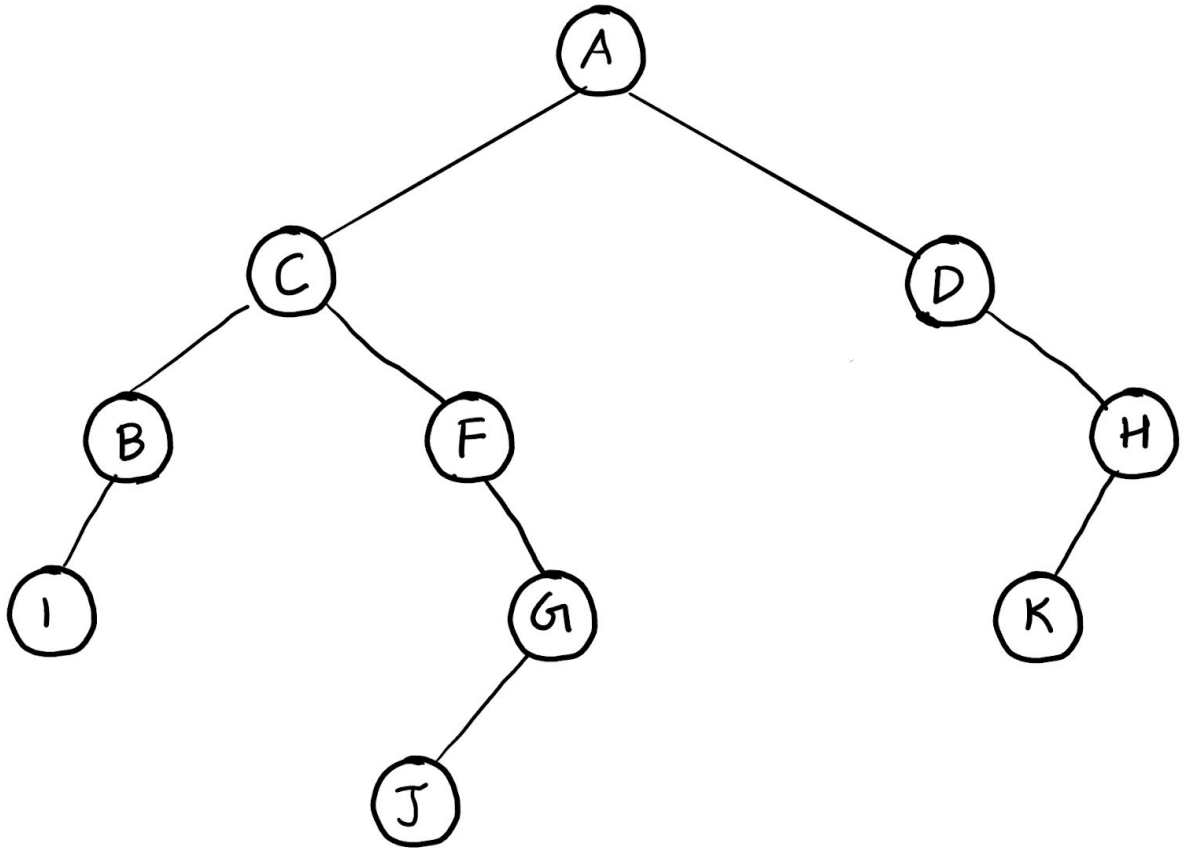
1. **(15 pts)** Given the following infix expression: (a – b + c * d) – e * f / (h + i). Produce a binary expression tree. Recall, leaves of the tree are *operands*, and other internal nodes are the *operators*.



---

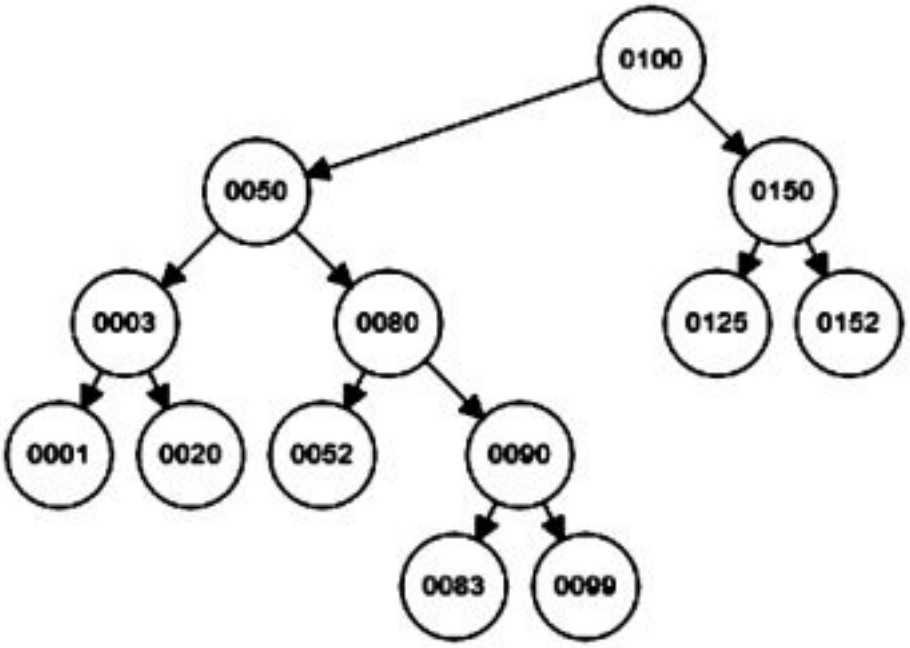
2. **(15 pts)** Given the following pre-order and in-order traversals, reconstruct the appropriate binary tree. **NOTE: You must draw a single tree that works for both traversals**.

   Pre-order: A, C, B, I, F, G, J, D, H, K
   In-order: I, B, C, F, J, G, A, D, K, H



---

3. (30 pts) Given the following binary tree (where nullptr height == -1):



a. (3 pts) What is the *height* of the tree?
Answer : Height of the tree is 4

b. (3 pts) What is the *depth* of the *root* node?
Answer : Depth of root node is 0

c. (3 pts) At which level is the *root* node?
Answer : Level of root node is 0.

d. (3 pts) What is the *depth* of node 0020?
Answer : Depth of node 20 is 3

e. (3 pts) List the values of all leaf nodes.
Answer : 0001,0020,0052, 0083, 0099, 00125, 0152

f. (3 pts) What is the *height* of node 0020?
Answer : The node 0020 is a node. Therefore its height is 0.

g. (12 pts – 4 pts/traversal) Give the pre-order, in-order, and post-order traversals of this tree.
Answer :
Preorder : 100, 50, 3, 1, 20, 80, 52, 90, 83, 99, 150, 125, 152
Post order : 1, 20, 3, 52, 83, 99, 90, 80, 52, 125, 152, 150, 100

---

4. a. (5 pts) What is an AVL tree? Explain.

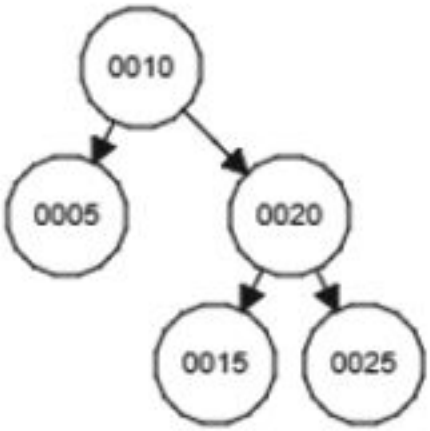
AVL are BST's with a special property that the difference between heights of left and right subtrees for a node is at maximum 1. It is a self balancing BST.

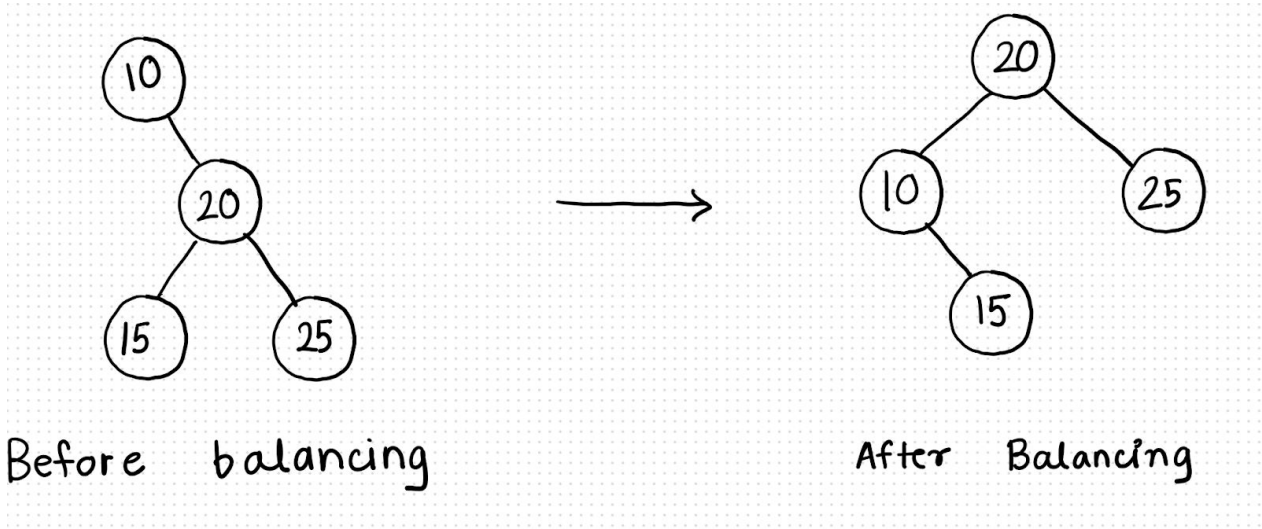
b. (5 pts) What is the purpose of an AVL tree? Explain.

AVL trees always have their height as O(log n) where n is the number of nodes. In BST's a situation may arise such that we get a skewed tree and the complexity of operations increases to O(n). In AVL trees since height is O(log n), we know that the complexity of operations has an upper bound of O(log n). This makes AVL trees more reliable than binary search trees

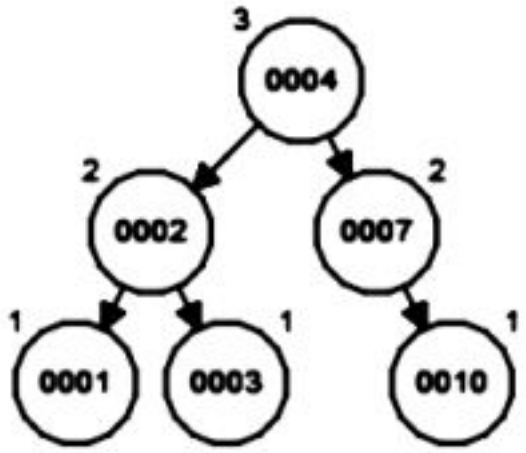
---

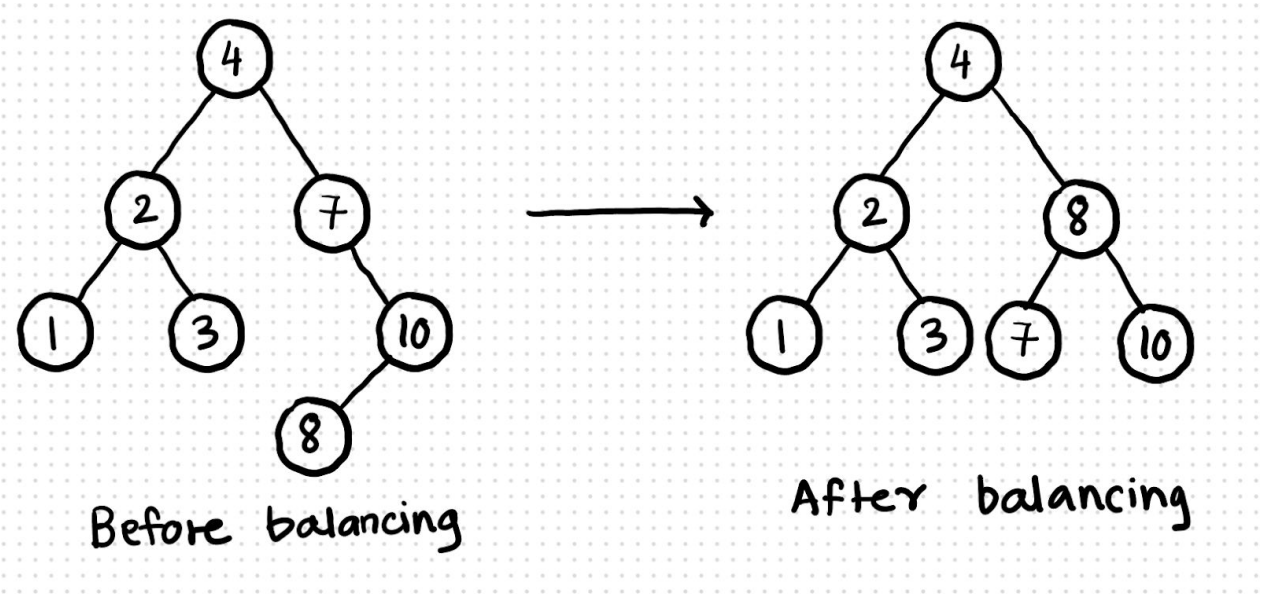
5. (10 pts) Remove 0005 from the following AVL tree; draw the resulting tree:



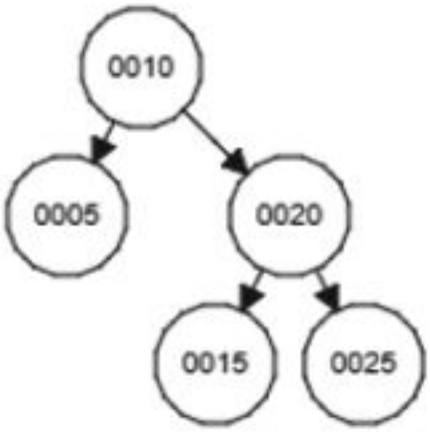


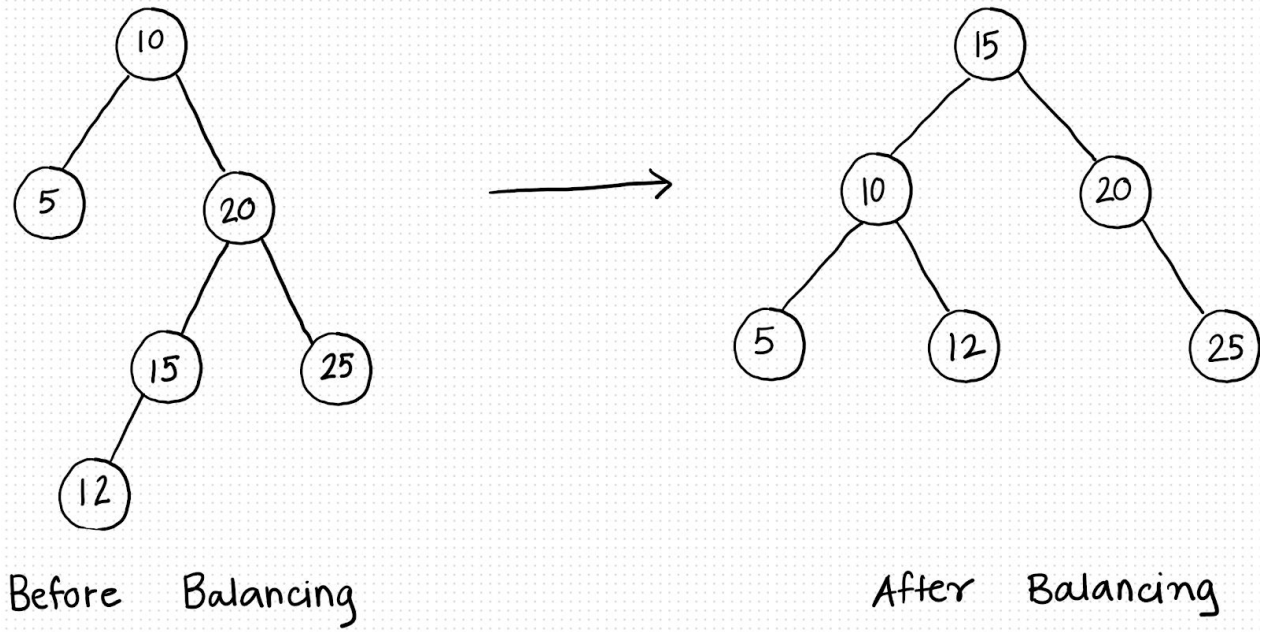
6. (10 pts) Insert the value 0008 into the following AVL tree; draw the resulting tree:





7. (10 pts) Insert the value 0012 into the following AVL tree; draw the resulting tree:




10
5
20
15
25
12
15
10
20
5
12
25
Before Balancing
After Balancing