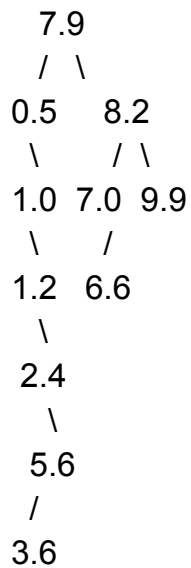(text) Type of Tree [10 points] Given each of the following arrays, create a BST by adding each element
of the array in the order it appears in the array. Afterwards, indicate what is the height of the resulting
Tree.

a) [7.9, 0.5, 1.0, 6.5, 8.2, 7.0, 6.6, 9.9,1.2, 2.4,5.6, 3.6]

```
      7.9
      / \
   0.5    8.2
     \     / \
    1.0  7.0  9.9
      \    /
     1.2  6.6
       \
      2.4
        \
       5.6
       /
     3.6
```
(sorry I don't have paper)

b) ["Petit Four", "Cupcake", "Donut", "Eclair", "Froyo", "Gingerbread", "Honeycomb"]

```
"Petit Four"
  /
"Cupcake"
    \
   "Donut"
      \
     "Eclair"
        \
       "Froyo"
          \
         "Gingerbread"
            \
           "Honeycomb"
```
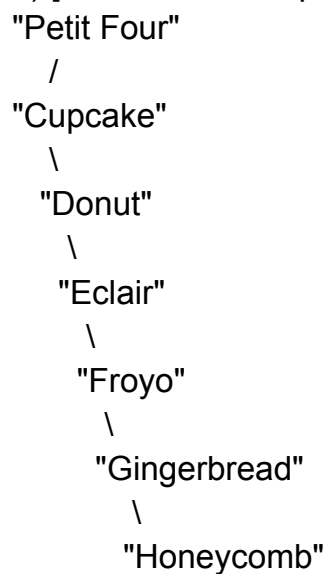
c) [32, 5, 94, 87, 10, 18, 85, 47, 25, 29]

```
      32
```

```
   /      \
  5        94
   \      /
   10   87
    \    /
   18 85
    \   /
   25 47
    \
    29
```

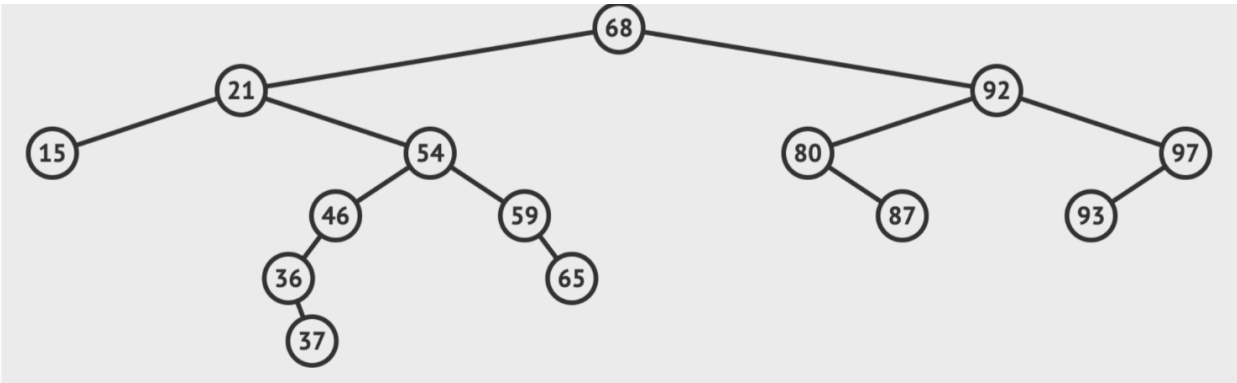d) [34, 30, 75, 77, 96, 48, 39, 50, 93, 13, 10, 5, 11, 20, 19]

```
        34
       /  \
     30     75
     /      /  \
    13     48  77
   /  \    / \   \
  10  20  39 50  96
 / \    /         /
5  11  19        93
```

2 (text) BST Traversal [10 points] Given the following BST, in which the data on an empty node is 0.

a. [6 points] What will be the resulting tree after doing preorder and inorder traversal and applying
the following operation on each node. Note round up the node values.
node.data += left.data + (right.data*2)
preorder:
68, 21, 15, 54, 46, 36, 37, 59, 65, 92, 80, 87, 97, 93
273, 144, 15, 218, 82, 110, 37, 189, 65, 366, 254, 87, 190, 93
Inorder:
15, 21, 36, 37, 46, 54, 59, 65, 68, 80, 87, 92, 93, 97
15, 144, 110, 37, 82, 218, 189, 65, 273, 254, 87, 366, 93, 190

b. [2 points] Are the resulting trees BSTs?
No, everything on the right should be greater than the new leaf, but 87 and 93 stay the same because they are leaves

c. [2 points] Are the resulting trees AVLs?
No, because it's not a BST

you can check the zybooks to see I did the labs