

## CSCI 204 Binary Tree Implementation Activity

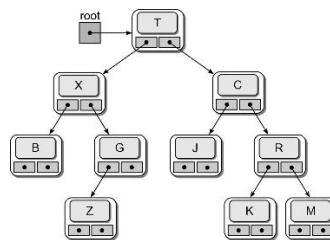
Student Name(s) \_\_\_\_\_

- We learned some basics of binary trees. Most of the operations that can be applied to a binary tree use some kind of recursion, due to the structure of binary trees. In this exercise, you are asked to write a recursive function to compute the height of a binary tree. The height of a binary tree is recursively defined by the following relation.

$$\text{height}(\text{node}) = 1 + \text{maximum}(\text{ height of left child of the node, height of right child of the node})$$

Write the function for the class BinTree that was discussed in the class. The current implementation of this class BinTree is on the course website. You will just need to add the height() function to the class.

After finishing the height function, add a call to this function and print out the result in the “testbintree.py” program. The height of the tree built manually in the test program should be 3. See the picture below.



- Rewrite the “testbintree.py” function such that the steps that build the above tree are put into a function “build\_tree1” that returns the tree built. That is, instead of having the code in lines 5 through 34 in the “main()” function, move all these lines into the function “build\_tree1” and have that function return the tree to the main function,

bintree = built\_tree1() # this should replace the code from line 5 through line 34

- Following a similar pattern to Problem 2, write another function called “built\_tree2()” that returns the tree pictured in c) of the following diagram. I borrowed the diagram from the textbook. The picture below has 3 trees, we just need to build. Then call the height function to compute and print the height, which should be 7.

