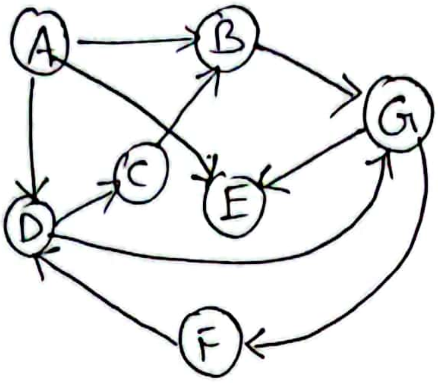
```
# Task - 1
class Node:
    def __init__(self, val=None, left=None, right=None):
        self.data = val
        self.left = left
        self.right = right


def get_height(s):
    if s is not None:
        return 1 + max(get_height(s.left), get_height(s.right))
    return 0


s = Node(15)
s.left = Node(10)
s.right = Node(20)
s.left.left = Node(8)
s.left.right = Node(12)
s.right.left = Node(16)
s.right.right = Node(25)

print("Height: ", get_height(s))


# Task - 2
class Node_New:

    def __init__(self, data):
        self.data = data
        self.left = None
        self.right = None



def get_lvl(node, data, i):
    if node is not None:
        if node.data != data:
            lvl_right = get_lvl(node.left, data, i + 1)
            if lvl_right != 0:
                return lvl_right

            lvl_left = get_lvl(node.right, data, i + 1)
        else:
            return i
        return lvl_right + lvl_left

    return 0



def Level(node, data):
    return get_lvl(node, data, 1)



s = Node_New(10)
s.left = Node_New(20)
s.right = Node_New(30)
s.left.left = Node_New(11)
s.left.right = Node_New(12)
```

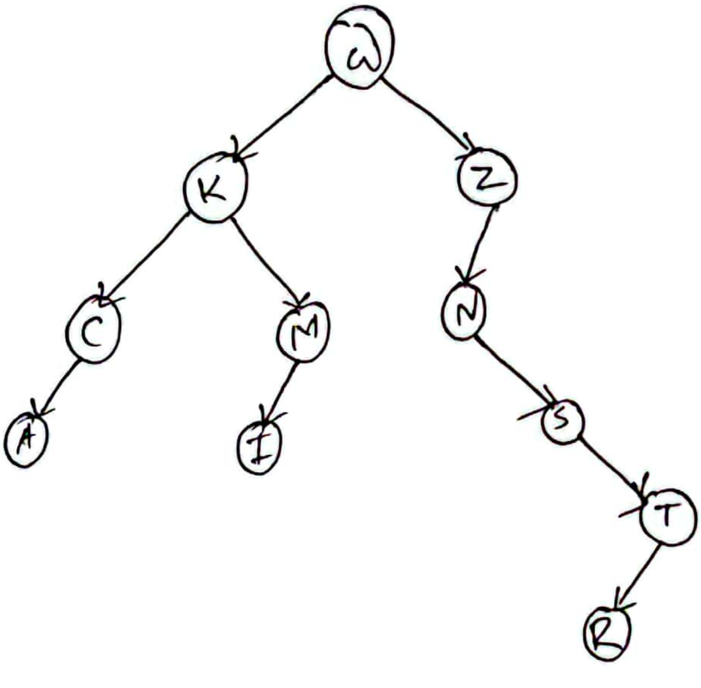
```
print('Level : ', Level(s, 20))


# Task - 3,4,5
class Node:
    def __init__(self, val):
        self.left = None
        self.right = None
        self.val = val

def printInorder(s):
    if not s:
        return
    printInorder(s.left)
    print(s.val, end=",")
    printInorder(s.right)


def postorder_traversal(s):
    if not s:
        return
    postorder_traversal(s.left)
    postorder_traversal(s.right)
    print(s.val, end=",")


def preorder_traversal(s):
    if not s:
        return
    print(s.val, end=",")

    preorder_traversal(s.left)

    preorder_traversal(s.right)



s = Node(1)

s.left = Node(2)

s.right = Node(3)

s.left.left = Node(4)

s.left.right = Node(5)

print("Preorder traversal :")

print(preorder_traversal(s))


print("Inorder traversal :")

print(printInorder(s))


print("Postorder traversal :")

print(postorder_traversal(s), )
```

Tahsin Ashrafee Susmit
20301088
Sec: 07

Task-6



Task-7

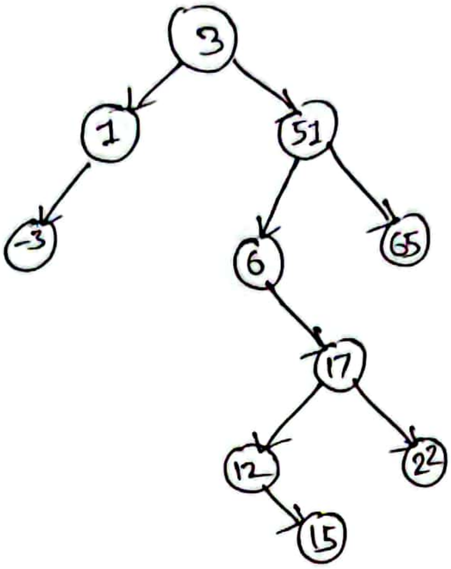


Preorder : w, K, C, A, M, I, Z, N, S, T, R

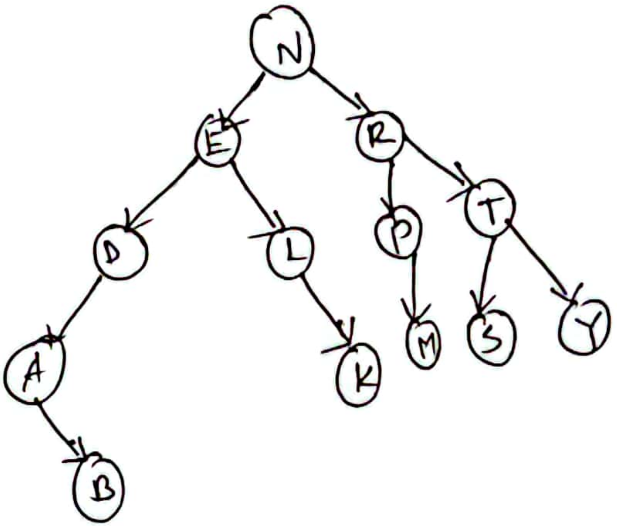
In-order : A, C, K, I, M, w, N, S, R, T, Z

post-order : A, C, I, M, K, R, T, S, N, Z, w

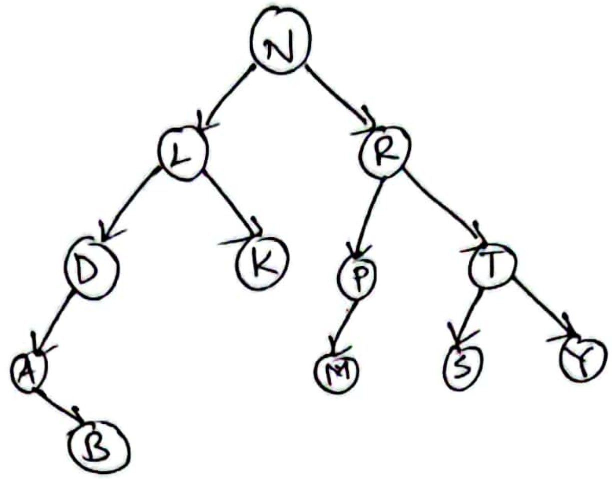
(a) BST :



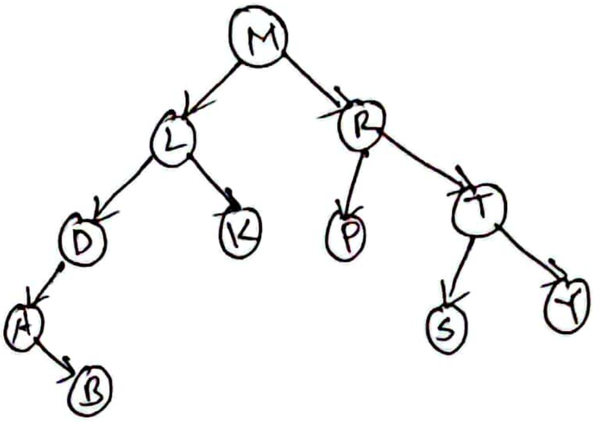
(b)



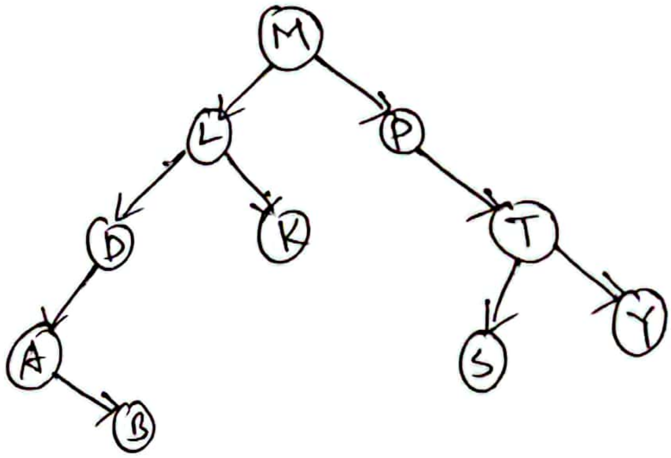
step 1 :



step 2 :



step 3 :


M
L
P
D
K
T
A
S
Y
B