Assignment #6
Course: CPSC 442 (Python for Data Science)

Please Note: You need to use a *try-except* block wherever the need arises for any of the following questions. Do not use Tree, Linked List related libraries.

1. Your task is to re-create a binary tree using a pre-order and in-order traversal lists provided. You may need to create a class structure for Tree.
   In the end, you need to print the tree content using post-order tree traversal mechanism.
   Those not aware of binary tree and tree traversal may read it using the following hyperlink.
   Binary Tree Tree Traversal

```
Example:

Inorder traversal: [4, 2, 1, 5, 3, 6]
Preorder traversal: [1, 2, 4, 3, 5, 6]

Solution:
    1
   / \
  2   3
 /   / \
4   5   6

Post-order traversal: 4,2,5,6,3,1
```

2. You are given a 2-D list. Every list has a dish and the ingredients used in it. Your task is to group the dishes by ingredients. **Include the dish only if it is made by more than 1 ingredient.**
   Also, the output should be in alphabetical order of the ingredients.
   You need to write a function for this task.

```
Example:

  Input  = [["Salad", "Tomato", "Cucumber", "Salad", "Sauce"],
            ["Pizza", "Tomato", "Sausage", "Sauce", "Dough"],
            ["Quesadilla", "Chicken", "Cheese", "Sauce"],
            ["Sandwich", "Salad", "Bread", "Tomato", "Cheese"]]

Solution:
    Output = [["Cheese", "Quesadilla", "Sandwich"],
              ["Salad", "Salad", "Sandwich"],
              ["Sauce", "Pizza", "Quesadilla", "Salad"],
              ["Tomato", "Pizza", "Salad", "Sandwich"]]

Explanation: 1st element of input list i.e. Salad, Pizza, Quesadilla, Sandwich are
names of dishes. The remaining elements are the ingredients used.
```

In the output, 1st element is the ingredient name and the remaining elements are the dishes made from them. Also note that the output is sorted by names of the ingredients.

3. You are given an array of string and an array of patterns. **Your task is to check that for every i and j, the strings and patterns should be the same** i.e.
   strings[i] = strings[j] and patterns[i] = patterns[j]
   Write a function to return True or False.

```
Example:

strings = ["Python", "Programming", "Programming"] and patterns = ["x", "y", "y"]
Solution: True

strings = ["Python", "Programming", "Class", “Programming”] and patterns = ["x", "y",
"y",”y”]
Solution: False


Explanation:
In 1st example, strings[2]=strings[3] and patterns[2]=patterns[3] => True
In 2nd example, strings[2] != strings[3] != strings[4] and
               patterns[2] = patterns[3] = patterns[4]    => False
```

4. Your task is to create a binary Tree of digits from 0 to 9 using user inputs. Then display the tree using in-order traversal.
   Next, your task is to iterate through the tree from the root to the leaf nodes and print the path. You also need to sum up all the paths in it, as shown in the example.
   Create appropriate class and functions.

```
Example:  User Inputs #
    1
   / \
  0   4
 / \
3   1

Inorder traversal: 3,0,1,1,4

Paths: 1 -> 0 -> 3
       1 -> 0 -> 1
       1 -> 4

Sum of all paths: 103 + 101 + 14 = 218

Explanation: Path traced would be from root to leaf nodes. Sum is the additions of
all paths.
```

Hint: for user input, ask for root, left and right node values respectively. If no value needs to be assigned, you can use None.