'''

--> PROBLEM STATEMENT :

        Matrix = 3X3

        Sum = 15

        Digits = 1 to 9

--> APPROACH :

        ---> Combinations having sum 15

        ---> Unique Combinations having sum 15 (Unique Digits)

        ---> No. of combinations from each digits

        ---> Center element is the digit which have 4 combinations

        ---> Corner elements are the digits which have 3 combinations

        ---> Center elements in the corner row or column are the digits which have 2 combinations

--> PATTERN :

            2  7  6

            9  5  1

            4  3  8

'''

# Given Data in Problem Statement

print()

digits = [1,2,3,4,5,6,7,8,9]

digits.sort()

num = 15

combinations = []

for i in range(len(digits)):

    temp\_list1 = []

    for j in range(i+1,len(digits)):

        temp\_list2 = []

        for k in range(j+1,len(digits)):

            if int(digits[i])+int(digits[j])+int(digits[k]) == 15:

                temp\_list2.append(digits[i])

                temp\_list2.append(digits[j])

                temp\_list2.append(digits[k])

        combinations.append(temp\_list2)

# print("Combinations : ",combinations)

# Remove empty lists using filter()

combinations = list(filter(None, combinations))

print("Unique Combinations (Sum=15): ",combinations)

# item at 0 index in the sublist is digit and the item at 1 index of sublist is the count of that digit occurred in the unique combinations

digit\_count\_combination = []

for i in digits:

    temp\_list = []

    count = 0

    temp\_list.append(i)

    for j in combinations:

        if i in j:

            count += 1

    temp\_list.append(count)

digit\_count\_combination.append(temp\_list)

# item at 0 index in the sublist is digit and the item at 1 index of sublist is the count of that digit occurred in the unique combinations

print("Digits Occurance in Combinations : ",digit\_count\_combination)

# Sorting digit\_count\_combination list onn the basis of their digit's count

sorted\_count = sorted(digit\_count\_combination,key= lambda x:x[1])

print("Sorted Count : ",sorted\_count)

# Center Digit : Digit occured in 4 different combinations

# Corner Digits : Digit occured in 3 different combinations

# Middle Digits : Digit occured in 2 different combinations

# Finding digits with their position

center\_digit = []

corner\_digits = []

middle\_digits = []

for i in sorted\_count:

    if i[1] == 4:

        center\_digit.append(i[0])

    elif(i[1]==3):

        corner\_digits.append(i[0])

    else:

        middle\_digits.append(i[0])

print()

print("Center Digit : ",center\_digit)

print("Corner Digits : ",corner\_digits)

print("Middle Digits : ",middle\_digits)

middle\_digits.reverse()

matrix = []

# print(matrix)

center\_index = 0

corner\_index = 0

middle\_index = 0

for i in range(9):

    if i%2==0:

        if i==4:

            matrix.append(center\_digit[center\_index])

            center\_index += 1

        else:

            matrix.append(corner\_digits[corner\_index])

            corner\_index += 1

    else:

        matrix.append(middle\_digits[middle\_index])

        middle\_index += 1

# Printing Matrix

print()

print("Matrix List : ",matrix)

print("Matrix : ")

matrix\_index = 0

for i in range(3):

    for j in range(3):

        print(matrix[matrix\_index],end=" ")

        matrix\_index += 1

print()

print("In this Matrix : Sum(rows\_elements = Sum(columns\_elements) = Sum(diagonal\_elements = 15))")

print()