Sort the given matrix

```
SIZE = 10
def sortMat(mat, n) :
    temp = [0] * (n * n)
    k = 0
    for i in range(0, n) :

        for j in range(0, n) :

            temp[k] = mat[i][j]
            k += 1
    temp.sort()
    k = 0

    for i in range(0, n) :

        for j in range(0, n) :
            mat[i][j] = temp[k]
            k += 1

def printMat(mat, n) :

    for i in range(0, n) :

        for j in range( 0, n ) :

            print(mat[i][j] , end = " ")

        print()
mat = [ [ 5, 4, 7 ],
        [ 1, 3, 8 ],
        [ 2, 9, 6 ] ]
n = 3

print( "Original Matrix:")
printMat(mat, n)
```

```
sortMat(mat, n)

print("\nMatrix After Sorting:")
printMat(mat, n)
```

Check if all rows of a matrix are circular rotations of each other

```
MAX = 1000
def isPermutedMatrix(mat, n) :
    str_cat = ""
    for i in range(n) :
        str_cat = str_cat + "-" + str(mat[0][i])
    str_cat = str_cat + str_cat
    for i in range(1, n) :
        curr_str = ""

        for j in range(n) :
            curr_str = curr_str + "-" + str(mat[i][j])

        if (str_cat.find(curr_str)) :
            return True

    return False

# Driver code
if __name__ == "__main__" :
    n = 4
    mat = [[1, 2, 3, 4],
        [4, 1, 2, 3],
        [3, 4, 1, 2],
        [2, 3, 4, 1]]
```

```
if (isPermutedMatrix(mat, n)):
    print("Yes")
else :
    print("No")
```