# -\*- coding: utf-8 -\*-

"""

Created on Tue Mar 3 16:42:56 2020

@author: sebas

"""

# Function to rotate the matrix

# 90 degree clockwise

def rotate90Clockwise(A):

N = len(A[0])

for i in range(N // 2):

for j in range(i, N - i - 1):

temp = A[i][j]

A[i][j] = A[N - 1 - j][i]

A[N - 1 - j][i] = A[N - 1 - i][N - 1 - j]

A[N - 1 - i][N - 1 - j] = A[j][N - 1 - i]

A[j][N - 1 - i] = temp

# Driver code

import numpy as np

import matplotlib.pyplot as plt

a=np.array([[1,1,1,1,1,1,0,1,1,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1,1,1,1,1,1,1,1,1,1],

[1,1,1,1,1,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,1,1,1,1,1,1,1,1,1],

[1,1,1,1,1,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,1,1,1,1,1,1,1,1,1],

[1,1,1,1,1,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,1,1,1,1,1,1,1,1,1],

[1,1,1,1,1,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,1,1,1,1,1,1,1,1],

[1,1,1,1,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1],

[1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1],

[1,1,1,1,1,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,0,1,1,0,0,1,1,1,1,1],

[1,1,1,1,1,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,0,0,0,1,1,1,0,0,1,1,1,1],

[1,1,1,1,1,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,0,0,1,1,1,1,0,0,1,1,1],

[1,1,1,1,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,0,0,1,1,1,1,1,0,1,1,1],

[1,1,1,1,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1,1,1,1,1,1,0,1,1],

[1,1,1,1,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1,1,1,0,0,1,0,0,1],

[1,1,1,1,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,0,0,0,1,1,1,0,1,1,1,0,0,0,1,0,1],

[1,1,1,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,0,1,1,1,0,1,1,1,0,0,0,1,0,0],

[1,1,1,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,0,1,1,1,0,0,1,1,0,0,0,1,0,0],

[1,1,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,0,1,1,1,0,0,1,1,0,0,0,1,0,0],

[1,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,0,1,1,1,0,0,1,1,1,0,0,1,1,0],

[1,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,1,1,1,0,0,0,1,1,1,1,1,1,0],

[0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,0,1,1,1,1,0,0,1,0,1,1,1,1,1,1,0],

[1,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,0,1,1,0,0,0,0,0,1,0,0],

[1,1,0,0,0,0,0,1,1,1,0,0,0,1,1,1,1,1,1,1,1,1,1,1,0,0,1,0,0,0,0,0,0,1,0],

[1,1,1,0,0,0,1,1,1,1,1,1,0,0,0,1,1,1,1,1,1,1,0,0,1,1,0,0,0,0,0,0,1,0,0],

[1,1,0,0,0,0,1,1,1,0,0,0,1,1,0,0,0,0,0,0,0,0,0,1,1,1,0,0,0,0,0,0,0,0,0],

[1,0,0,0,0,0,1,1,1,0,0,0,1,1,1,1,1,0,0,0,0,1,1,1,1,1,1,0,0,0,0,0,0,0,0],

[1,1,1,1,0,0,1,1,1,1,1,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1,0,1],

[1,1,1,1,1,0,0,1,1,1,1,1,1,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,1,0,0,1],

[1,1,1,1,1,0,0,1,1,1,1,1,1,0,0,0,0,0,0,0,1,1,1,1,0,0,0,0,0,0,1,1,0,1,1],

[1,1,1,1,1,1,0,0,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,1,1],

[1,1,1,1,1,1,1,0,0,1,1,1,1,1,0,0,0,1,1,1,0,0,0,0,0,0,0,0,0,1,0,0,1,1,1],

[1,1,1,1,1,1,1,1,0,0,1,1,1,1,1,0,1,1,1,1,1,1,0,1,1,1,0,0,1,0,0,1,1,1,1],

[1,1,1,1,1,1,1,1,1,0,0,1,1,1,1,1,0,0,1,1,1,1,1,1,1,0,0,1,0,0,1,1,1,1,1],

[1,1,1,1,1,1,1,1,1,1,0,0,0,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1],

[1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,1,1,0,1,1,0,0,0,0,0,1,1,1,1,1,1,1,1],

[1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1]])

rotate90Clockwise(a)

printMatrix(a)

#https://www.geeksforgeeks.org/rotate-matrix-elements/

fig,ax=plt.subplots()

im=ax.imshow(a)

# Python program to illustrate reflection

def reverse(seq):

SeqType = type(seq)

emptySeq = SeqType() #

if seq == emptySeq:

return emptySeq

restrev = reverse(seq[64:])

first = seq[64:64]

# Combine the result

result = restrev + first

return result

# Driver code

print(reverse([[1,1,1,1,1,1,0,1,1,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1,1,1,1,1,1,1,1,1,1],

[1,1,1,1,1,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,1,1,1,1,1,1,1,1,1],

[1,1,1,1,1,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,1,1,1,1,1,1,1,1,1],

[1,1,1,1,1,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,1,1,1,1,1,1,1,1,1],

[1,1,1,1,1,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,1,1,1,1,1,1,1,1],

[1,1,1,1,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1],

[1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1],

[1,1,1,1,1,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,0,1,1,0,0,1,1,1,1,1],

[1,1,1,1,1,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,0,0,0,1,1,1,0,0,1,1,1,1],

[1,1,1,1,1,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,0,0,1,1,1,1,0,0,1,1,1],

[1,1,1,1,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,0,0,1,1,1,1,1,0,1,1,1],

[1,1,1,1,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1,1,1,1,1,1,0,1,1],

[1,1,1,1,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1,1,1,0,0,1,0,0,1],

[1,1,1,1,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,0,0,0,1,1,1,0,1,1,1,0,0,0,1,0,1],

[1,1,1,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,0,1,1,1,0,1,1,1,0,0,0,1,0,0],

[1,1,1,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,0,1,1,1,0,0,1,1,0,0,0,1,0,0],

[1,1,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,0,1,1,1,0,0,1,1,0,0,0,1,0,0],

[1,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,0,1,1,1,0,0,1,1,1,0,0,1,1,0],

[1,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,1,1,1,0,0,0,1,1,1,1,1,1,0],

[0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,0,1,1,1,1,0,0,1,0,1,1,1,1,1,1,0],

[1,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,0,1,1,0,0,0,0,0,1,0,0],

[1,1,0,0,0,0,0,1,1,1,0,0,0,1,1,1,1,1,1,1,1,1,1,1,0,0,1,0,0,0,0,0,0,1,0],

[1,1,1,0,0,0,1,1,1,1,1,1,0,0,0,1,1,1,1,1,1,1,0,0,1,1,0,0,0,0,0,0,1,0,0],

[1,1,0,0,0,0,1,1,1,0,0,0,1,1,0,0,0,0,0,0,0,0,0,1,1,1,0,0,0,0,0,0,0,0,0],

[1,0,0,0,0,0,1,1,1,0,0,0,1,1,1,1,1,0,0,0,0,1,1,1,1,1,1,0,0,0,0,0,0,0,0],

[1,1,1,1,0,0,1,1,1,1,1,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1,0,1],

[1,1,1,1,1,0,0,1,1,1,1,1,1,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,1,0,0,1],

[1,1,1,1,1,0,0,1,1,1,1,1,1,0,0,0,0,0,0,0,1,1,1,1,0,0,0,0,0,0,1,1,0,1,1],

[1,1,1,1,1,1,0,0,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,1,1],

[1,1,1,1,1,1,1,0,0,1,1,1,1,1,0,0,0,1,1,1,0,0,0,0,0,0,0,0,0,1,0,0,1,1,1],

[1,1,1,1,1,1,1,1,0,0,1,1,1,1,1,0,1,1,1,1,1,1,0,1,1,1,0,0,1,0,0,1,1,1,1],

[1,1,1,1,1,1,1,1,1,0,0,1,1,1,1,1,0,0,1,1,1,1,1,1,1,0,0,1,0,0,1,1,1,1,1],

[1,1,1,1,1,1,1,1,1,1,0,0,0,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1],

[1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,1,1,0,1,1,0,0,0,0,0,1,1,1,1,1,1,1,1],

[1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1]])