Computer Science I Bill Griffith PhD
HW8 18 pts **Due Sunday, November 10, 11:59 PM**

PART A – 12 points) Write a Python program that creates an n x n Magic Square, where n is an odd integer between 3 and 15, using the consecutive numbers from 1 to n squared, such that each of the rows, columns, and diagonals add up to the same number and print that number. Call the two-dimensional list (array) **Magic_Square. Use the "paper and pencil" algorithm we discussed in class.**
The program must also record and print the various counts associated with specific moves during the construction of the square. Lastly, **and only for the 3x3 Magic_Square**, the program must count how many iterations it will take to randomly create a Square that is a perfect duplicate of the 3x3 Magic_Square.

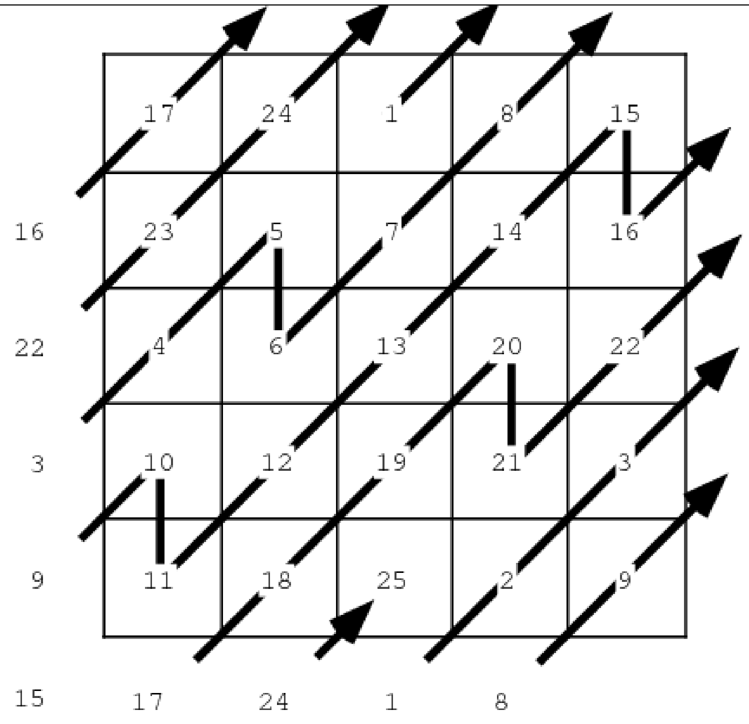
Sample Outputs:

```
This program creates an odd-sided Magic Square,
  where the sum of each row, column and diagonal
  adds up to the same number

The program will also record and print the various counts associated
with specific moves during the construction of the square

Lastly, and only for the 3x3 Magic_Square,
the program must count how many iterations it will take
to randomly create a Square that is a
perfect duplicate of the 3x3 Magic_Square.

Input size of an nxn odd-sided square: 5

  17  24   1   8  15
  23   5   7  14  16
   4   6  13  20  22
  10  12  19  21   3
  11  18  25   2   9

 The sum of each row, column, and diagonal is:  65
incount:  12
blocked:  3
row out:  4
col out:  4
both out:  1
```

```
Program End

This program creates an odd-sided Magic Square,
  where the sum of each row, column and diagonal
  adds up to the same number

The program will also record and print the various counts associated
with specific moves during the construction of the square

Lastly, and only for the 3x3 Magic_Square,
the program must count how many iterations it will take
to randomly create a Square that is a
perfect duplicate of the 3x3 Magic_Square.


Input size of  an nxn odd-sided square: 3

   8   1   6
   3   5   7
   4   9   2

 The sum of each row, column, and diagonal is:  15

incount:  2
blocked:  1
row out:  2
col out:  2
both out:  1

 Random Match for 3x3

[7, 8, 5, 6, 1, 9, 2, 3, 4]
100000
[8, 6, 4, 5, 3, 7, 1, 9, 2]
200000

[8, 1, 6, 3, 5, 7, 4, 9, 2]
It took  273066 tries to create a random square to match the Magic Square

 Program End
```

PART B – 6 points) Write another solution to the magic square problem using the mod operator to adjust the row/col indices. Just create the square with a new function. No counts required.

For Part A, the algorithm is given here:



Kraitchik (1942) gives general techniques of constructing even and odd squares of order n. For n odd, a very straightforward technique known as the Siamese method can be used, as illustrated above (Kraitchik 1942, pp. 148-149). It begins by placing a 1 in the center square of the top row, then incrementally placing subsequent numbers in the square one unit above and to the right. The counting is wrapped around, so that falling off the top returns on the bottom and falling off the right returns on the left. When a square is encountered that is already filled, the next number is instead placed *below* the previous one and the method continues as before. The method, also called de la Loubere's method, is purported to have been first reported in the West when de la Loubere returned to France after serving as ambassador to Siam.

Now, we are only doing odd-sided squares as in the example. This is the algorithm for Part A. So, you start in the middle of the top row and then proceed right one cell and then up one cell. Four situations can arise:

1) the cell is empty (that is, there is a 0 in the cell – when you set up the grid, you would load it with 0's with list comprehension), and you put the next number in the cell, or,

2) the cell already has something in it (or both row and column are out of range). In this case you return to the previous cell and drop down a row.

3) the row number is out of range but the column number is in range. In this case, you set the row number to the max (5 in the example) and place the next number in that cell.

4) the column number is out of range but the row number is not. The set the row number to the minimum (0) and place the next number there.