

**EECS 12 Assignment 5 (due 6pm, Nov. 23)**  
**Fall 2013**

**I. Magic Square**

In this assignment, you are to implement a program for displaying a magic  $n \times n$  square ( **$n$  is an odd number**). In the homework here, a magic square is an arrangement of integers ( $1 \sim n^2$ ) in a square grid, where the numbers in each row and each column all add up to the same number. You DO NOT need to consider the sums in the diagonals in this homework.

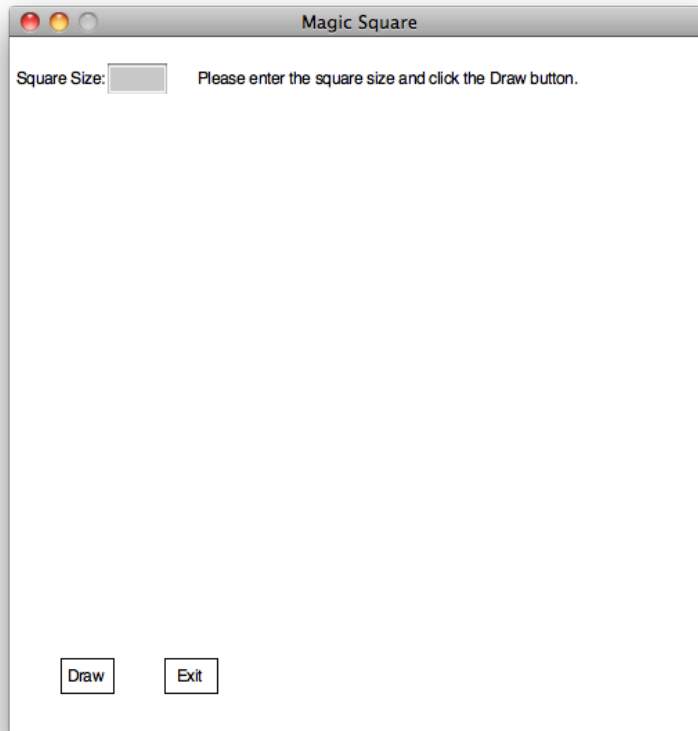
Here is the algorithm for construct a  $n \times n$  squares, it works only if  $n$  is odd.

```
Given any row, column as initial start place
For k = 1.....n*n
    Place k at [row][column]
    Increment row and column
    If the row or column is n, replace it with 0
    If the element at [row][column] has already been filled
        Set row and column to their previous values
        Decrement row
    If row is -1, replace it with n-1
```

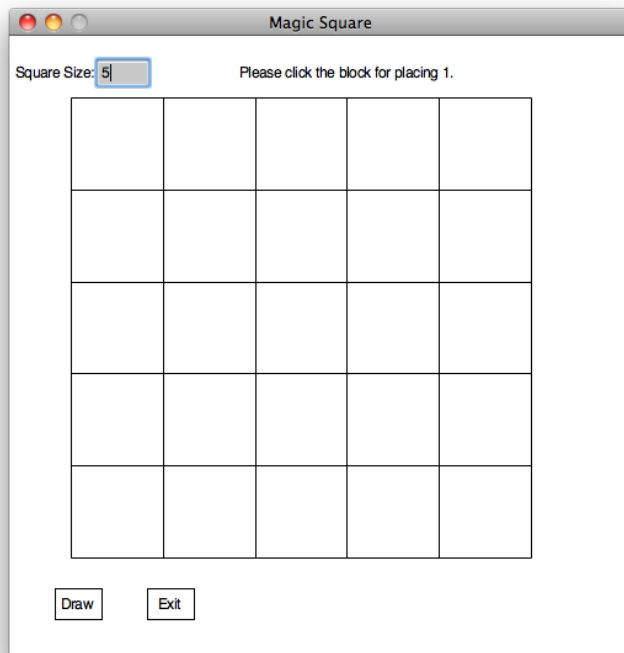
**II. The Program**

Here is an example of how your program should work.

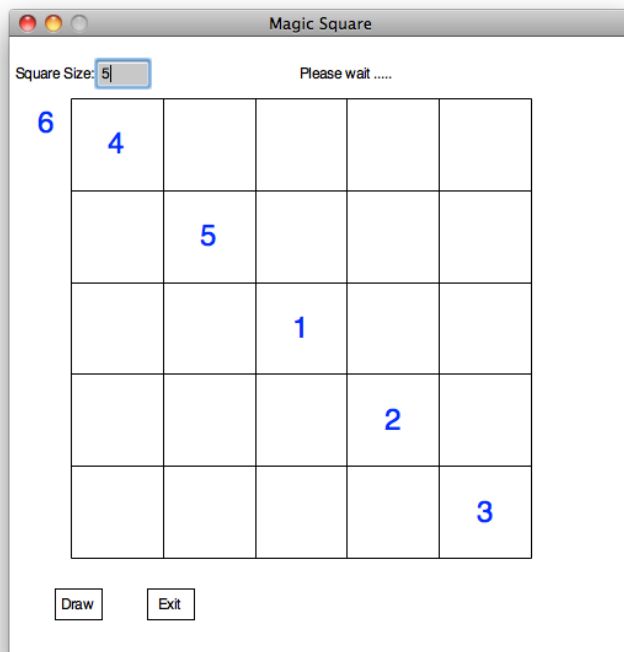
1. A 500\*500 window with title “Magic Square” should be created



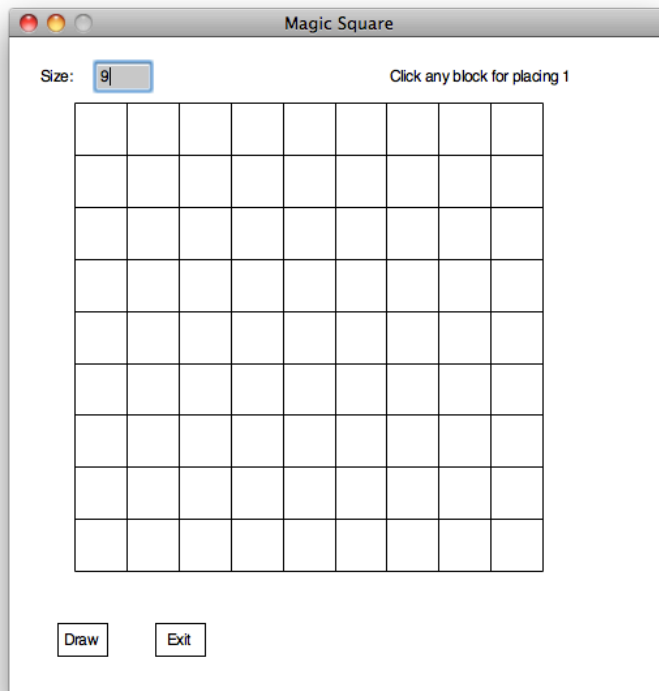
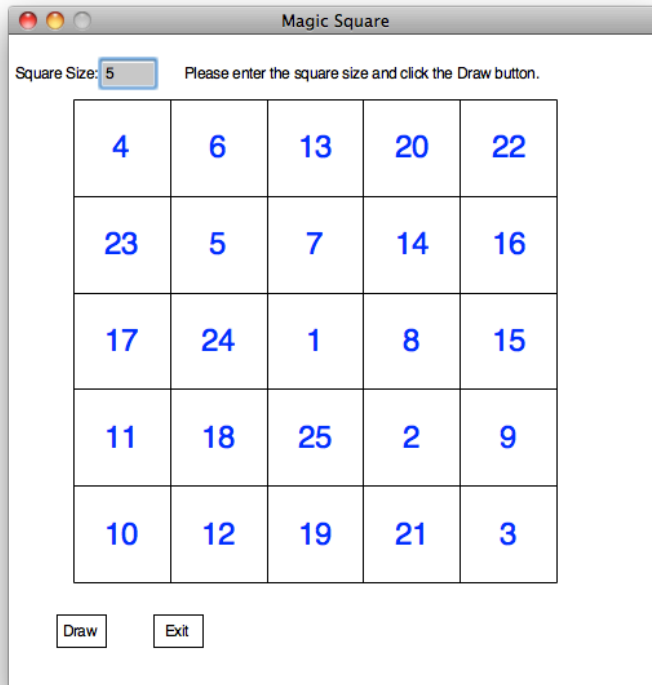
2. The user enters the desired dimension  $n$  in the text field and clicks the “Draw” button. A grid of  $n \times n$  will be shown. The program now waits for the user to click at one of the cells in the grid.



3. The user clicks at one of the cells. The program should start to display numbers from the upper left corner of the grid, and fly to their correct positions one by one. When in progress, program may not respond to any mouse click. (6 is shown where it first appears.)



4. After all numbers are displayed, the user may enter another size and click the “Draw” button again. The previous drawing is then cleared, and a new grid will be shown according to the new size set by the user.



5. If “Exit” is clicked, the program should exit and terminated.

### III. Using functions in magic square

1. You should use functions for animations (moving numbers to correct place)  
Here is an example of a function definition for move. You can also have different definitions as long as it works.

```
#function for moving obj from (startX, startY) to (endX, endY),  
# obj is a Text object. startX, startY, endX, endY are numbers  
def move(obj, startX, startY, endX, endY):
```

Another function is to convert logic position of a number in the matrix into coordinates in the window. For example, for a 5\*5 matrix, if we want to put 1 in the center box, it should be shown in the (2,2) cell and (250, 240) in the window. We can use this function to convert (2,2) into (250, 240) for display 1. Here is a definition for this conversion. You can have different definitions as long as it works.

```
#function for calculating actual coordinates in window given place in  
#matrix  
# left_border, top_border are numbers representing coordinates of the  
# two borders of the matrix. width is the width of one cell in the  
# matrix. Column and row represents the position in matrix that we want  
# to do the transformation  
def calculateCoordinates (left_border, top_border, width, column, row):
```

#### IV. Grading Criteria (90 points):

- Display a matrix of correct size using the user's size input (15pt)
- Produce a correct magic square (15pt)
- Construct the correctly magic square with 1 at the position where the user selected (10pt)
- Use functions for number animation and conversion(15pt)
- Draw button will clear existing square and draw a new square using the new size (15pt)
- Display good messages to user (10pt)
- Exit the program correctly when the Exit button is clicked (10pt)

#### Submission:

Submit your homework before **6pm, Nov. 23 (Saturday)** to the EEE Dropbox. Submit only the python source file, with file name "hw5.py".