

ECS 102 Baruch
Spring 2019
HW 8
Due: Thursday, April 4, 11:59 pm

For this assignment you will be writing 2 programs in files, each with a main.

1. Start each program with a comment like this:

```
# Official Name: Amanda Kitredge  
# Nickname: Mandy  
# email: ajkitred@syr.edu  
# Assignment: Assignment 3, problem 1.  
# Date: September 13, 2018  
# a brief description of the problem.
```

Replace the *italicised fields* with appropriate information for you and for this assignment.

- In the "Official Name" field write your name as it appears on University documents.
- "Nickname" is the name you would like to be called by the course staff. If you don't use a nickname you can leave this line out.

2. In addition, each program should have comments describing the main sections of the program. They should be specific to the problem you are solving. Under each of these comments should be a chunk of code implementing the comment.

Each function and class method should have a comment describing what it does, what the parameters are and represent, and assumptions made about the parameters.

3. Good readability:

- Use meaningful names for all variables.
- Use blank lines to delineate small portions of code that work together to accomplish a task.

4. Test your program. Run each program several times to see that they are working and that you don't get the same output each time.

A. RandomWalk.py

Set up a graphics window 600 by 600 to go from -5 to 5 in both the x and y directions (that is, the coordinates should be from -5 to 5).

Write a function

```
linepaper(win)
```

It should line the window from -5 to 5 in both the x and y directions (making graph paper.)

Test what you have done so far.

Write a function

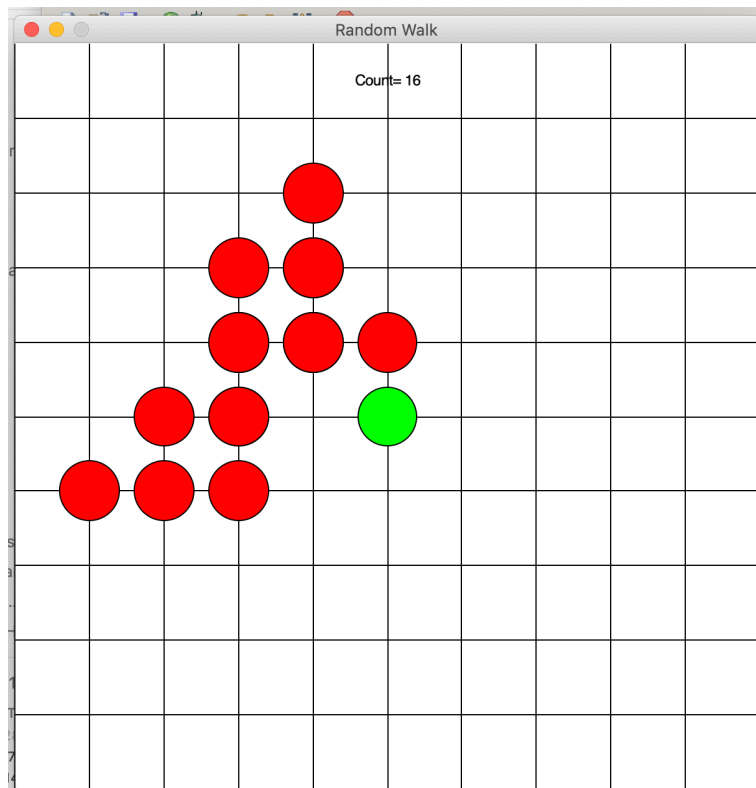
walk(win)

It should simulate a random walk as follows:

- Start at the origin. Draw a green circle there.
- repeat as long as the circle will be completely visible
 - take a step one unit up, down, left, or right
 - draw a red circle there
- print the number of steps taken (without going out of bounds)

Here is one sample output. Yours will come out very different from mine.

(You might not see a green circle if another, red circle is drawn at the same point. But you should see the green circle on some runs.)



Take a screenshot of one outcome and label it randomWalk.png (or .pdf)

B. ShuffleCards.py

In this assignment you will be shuffling cards in a list.

I will talk about **swapping** two values. In Python this can be done with simultaneous assignment:

```
a, b = b, a
```

Write a function **MakeDeck()** that creates a list of "cards" and returns it (the **deck**). For this assignment the cards can be simple, each just has a letter on it.

Write a function **Shuffle(deck)** that works as follows:

Take the first card in the list and swap it with a random card in the list.

Take the second card in the list and swap it with a random card in the part of the list going from the second card to the last card (inclusive)

Take the third card and swap it with a random card going from the third card to the end of the list. And so on. To see things are working properly, print the list as each pair is swapped.

The function should then return the shuffled deck.

main should create a deck (using the function), print it, shuffle the deck (using the function), and print it.

Once this is working, it would be nice to only show all the intermediate steps if the programmer wants them shown. Modify **shuffle** to take two parameters:

the **deck** and a Boolean variable **show** that can be **True** or **False**.

When the value of show is True, it prints all the intermediate steps. When the value of show is False, it doesn't print the intermediate steps.

Have two versions of main (call them main1 and main2) to call this new version of shuffle, once with True for the show argument and once with False for the show argument.

I give a sample output for each case, when show is True and when show is False.

Sample output with **show** True:

```
['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O']
['I', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'A', 'J', 'K', 'L', 'M', 'N', 'O']
['I', 'A', 'C', 'D', 'E', 'F', 'G', 'H', 'B', 'J', 'K', 'L', 'M', 'N', 'O']
['I', 'A', 'B', 'D', 'E', 'F', 'G', 'H', 'C', 'J', 'K', 'L', 'M', 'N', 'O']
['I', 'A', 'B', 'F', 'E', 'D', 'G', 'H', 'C', 'J', 'K', 'L', 'M', 'N', 'O']
['I', 'A', 'B', 'F', 'H', 'D', 'G', 'E', 'C', 'J', 'K', 'L', 'M', 'N', 'O']
['I', 'A', 'B', 'F', 'H', 'N', 'G', 'E', 'C', 'J', 'K', 'L', 'M', 'D', 'O']
['I', 'A', 'B', 'F', 'H', 'N', 'D', 'E', 'C', 'J', 'K', 'L', 'M', 'G', 'O']
```

```
[T, 'A', 'B', 'F', 'H', 'N', 'D', 'G', 'C', 'J', 'K', 'L', 'M', 'E', 'O']
[T, 'A', 'B', 'F', 'H', 'N', 'D', 'G', 'C', 'J', 'K', 'L', 'M', 'E', 'O']
[T, 'A', 'B', 'F', 'H', 'N', 'D', 'G', 'C', 'O', 'K', 'L', 'M', 'E', 'J']
[T, 'A', 'B', 'F', 'H', 'N', 'D', 'G', 'C', 'O', 'K', 'L', 'M', 'E', 'J']
[T, 'A', 'B', 'F', 'H', 'N', 'D', 'G', 'C', 'O', 'K', 'L', 'M', 'E', 'J']
[T, 'A', 'B', 'F', 'H', 'N', 'D', 'G', 'C', 'O', 'K', 'L', 'E', 'M', 'J']
[T, 'A', 'B', 'F', 'H', 'N', 'D', 'G', 'C', 'O', 'K', 'L', 'E', 'M', 'J']
[T, 'A', 'B', 'F', 'H', 'N', 'D', 'G', 'C', 'O', 'K', 'L', 'E', 'M', 'J']
[T, 'A', 'B', 'F', 'H', 'N', 'D', 'G', 'C', 'O', 'K', 'L', 'E', 'M', 'J']
```

(The last two lines are the same because it is printed in shuffle and it is printed in main.
Sometimes two lines are the same because a card is swapped with itself.)

Sample output with **show** False:

```
['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O']
['G', 'B', 'N', 'J', 'H', 'C', 'A', 'E', 'I', 'O', 'D', 'F', 'K', 'L', 'M']
```

Save two outputs from each version of main in a file shuffleOutput.txt

Upload:

- **RandomWalk.py** and **randomWalk.png** (or **pdf**)
- **ShuffleCards.py** and shuffleOutput.txt