CS 112 Lab 3 Random Walks

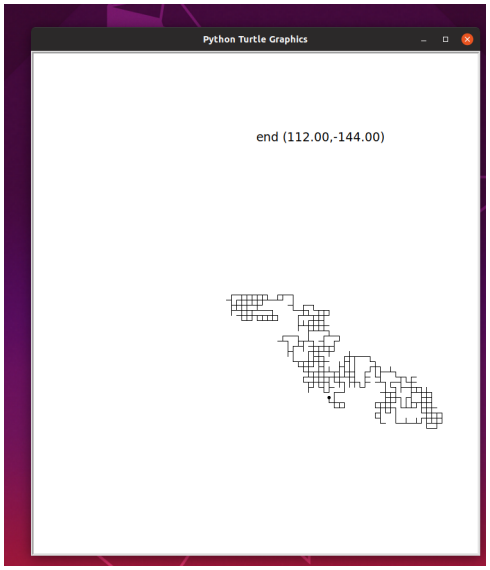
If your group has not yet submitted code for lab 2, then please do so by Mon. 16 Sep. at 10 AM.

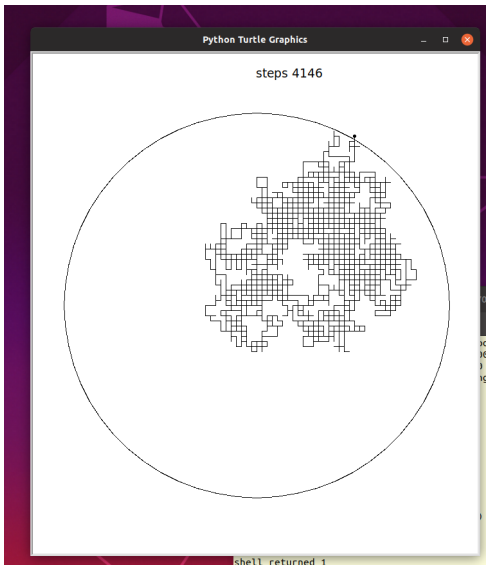
Problem. Design software to perform a random walk with graphics.

The overall objective of lab 3 is to improve your abiity to solve problems. In particular you will design functions that use a loop, either a for loop or a while loop to accomplish a particular task. These functions use turtle graphics.

A random walk starts at the origin and continues for a certain number N of steps. At each step, the walk moves one step at random, either north, east, south or west, each with probability 25%. Determine how far the ending point is from the origin after N steps. One walk of 1000 steps appears on the left in the following figure.

Another version of a random walk starts at the origin and counts the number of steps needed until the walk moves outside a circle centered at the origin with a particular radius. One such walk appears on the right in the following figure.





Each walk uses a step size of 8 pixels. Each walk starts at the origin.

You must design a function to draw a walk of a certain number of steps. It uses a for loop. You must also design a function that counts the number of steps a random walk needs to move outside a circle of a particular radius centered at the origin. It uses a while loop since the number of steps to move outside the circle is not known. You may also need to design other functions.

A random walk has applications in the real world. See Wikipedia's article.
`https://en.wikipedia.org/wiki/Random_walk`

Clearly a random walk requires the use of a function from the random module of Python. One function that I used in the design of a solver for a quadratic equation is randint(). It has 2 parameters low and high and returns an integer x with low $\leq$ x $\leq$ high. For lab 3, each step moves in one of four directions. So you can use randint(1, 4) to select a number between 1 and 4 inclusive. Then you must associate each number with a direction. The association is not obvious.

The random module has a better option for this lab called called choice(). It has a sequence parameter and returns a random element of the sequence. If I define a constant DIRECTIONS with ['n', 'e', 's', 'w'], a call to choice() with DIRECTIONS returns at random one of 'n', 'e', 's','w'.

Your design is due either at 10 AM on Wed. 18 Sep. if your lab day is Wed. or at 9 AM on Th. 19 Sep. if your lab day is Thursday.
I will hold an optional design session on Mon. 16 Sep. at 2 PM in CW 320.
Feel free to send me email or to see me in my office if you have questions about the design for this lab.