6.00 Quiz 2, 2011 ----------

Name

1. _____/15
2. _____/18
3. _____/16
4. _____/10
5. _____/19
6. _____/10
7. _____/12

Total _____ /100

This quiz is open book and open notes, but do not use a computer.

Please **write your name on the top of each page**. Answer all questions in the boxes provided.

1) Are each of the following True or False (15 points)

True — 1.1. In Python, classes **do not** enforce information hiding.

false — 1.2. In Python, classes **cannot** be used as a parameter of a function.

false — 1.3. Increasing the number of buckets in a hash table typically **increases** the amount of time needed to locate a value in the table.

True — 1.4. "Normal distribution" and "Gaussian distribution" are different names for the same thing.

false — 1.5. "Standard deviation" and "coefficient of variation" are different names for the same thing.

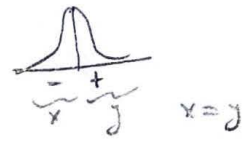
Name _____

2) Consider the following code:

```
import random
tots = [0.00]*3
maxVals = [0.0]*3
mean = 100.0
stdDevs = [0.0, 20.0, 40.0]
for i in range(1000):
    for j in range(len(tots)):
        next = random.gauss(mean, stdDevs[j])
        tots[j] += next
        if next > maxVals[j]:
            maxVals[j] = next
```

2.1. What are the expected values of each element of `tots` at the end of the code? Hint: `random.gauss(mu, sigma)` returns a random value chosen from a Gaussian distribution with mean `mu` and standard deviation `sigma`. (9 points)

tots = 1000 * 100
= 100000



Are each of the following True or False (9 points):

True — 2.2. One would expect `maxVals[0]` to be less than `maxVals[1]`.

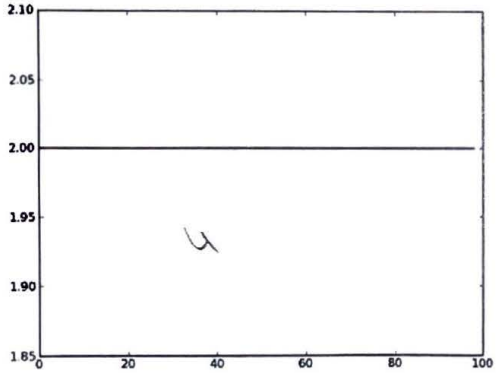
True — 2.3. One would expect `maxVals[1]` to be less than `maxVals[2]`.

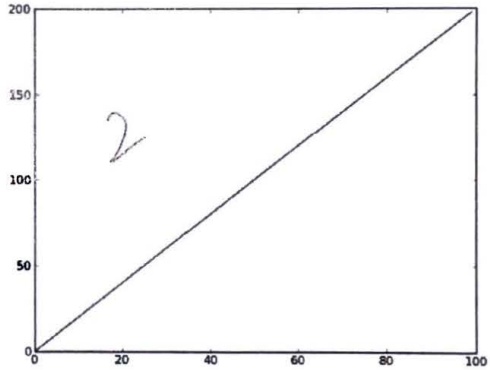
True — 2.3. If the code were run twice, the value of `tots[0]` would be the same each time.

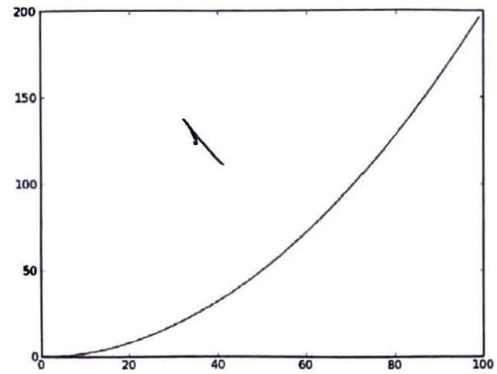
Name

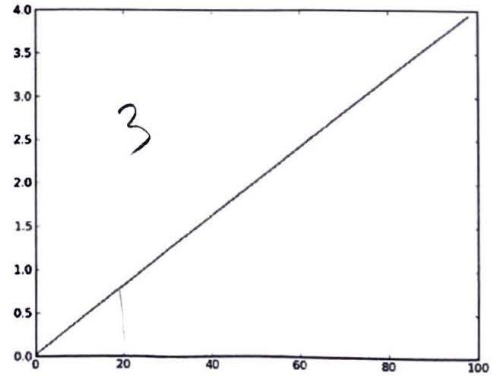
3) The code produces 4 plots. Match each plot below to a figure number by writing 1, 2, 3 or 4 **on the plots**. (16 points)

```
y1, y2, y3, y4 = [], [], [], []
for i in range(100):
    y1.append((i**2)/50.0)
    y2.append(2*i)
for i in range(99):
    y3.append(y1[i+1] - y1[i])
    y4.append(y2[i+1] - y2[i])
pylab.figure(1)
pylab.plot(y1)
pylab.figure(2)
pylab.plot(y2)
pylab.figure(3)
pylab.plot(y3)
pylab.figure(4)
pylab.plot(y4)
```









4) What does the following code print? (10 points)

```
class Shape(object):
    def __eq__(s1, s2):
        return s1.area() == s2.area()
    def __lt__(s1, s2):
        return s1.circum() < s2.circum()

class Rectangle(Shape):
    def __init__(self, h, w):
        self.height = float(h)
        self.width = float(w)
    def circum(self):
        return 2*(self.height + self.width)
    def __str__(self):
        return 'Rectangle with area ' + str(self.height*self.width)

class Square(Rectangle):
    def __init__(self, s):
        Rectangle.__init__(self, s, s)
    def __str__(self):
        return 'Square with side ' + str(self.height)

class Circle(Shape):
    def __init__(self, radius):
        self.radius = float(radius)
    def circum(self):
        return 3.14159*(2*self.radius)
    def __lt__(self, other):
        return self.radius < other.radius
    def __str__(self):
        return 'Circle with diameter ' + str(2.0*self.radius)

def reorder(L):
    for e in L:
        if e < L[0]:
            L[0] = e

L = [Square(6), Rectangle(2, 3), Circle(1)]
try:
    reorder(L)
    for e in L:
        print e
except:
    for e in L:
        print e
```

Rectangle with area 6.0
Rectangle with area 6.0
Circle with diameter 2.0

5) Write a function that uses a Monte Carlo simulation to find the probability of a run of at least 4 consecutive heads out of ten flips of a fair coin, and then returns that probability. Assume that 10,000 trials are sufficient to provide an accurate answer. You may call the function :

```
def simThrows(numFlips):
    """Simulates a sequence of numFlips coin flips, and returns True if
       the sequence contains a run of at least four consecutive
       heads and False otherwise."""
```

(19 points)

def sim(): #write your code below

```
cases = 0
for trial in range(10000):
    cases += simThrows(10)
result = float(cases)/10000
return result
```

6) If `pylab.polyfit` is used to fit an n^{th} degree and an $(n+1)^{th}$ degree polynomial to the same data, is one guaranteed to provide a least squares fit that is at least as good as the other? If so, which and why? If not, why not? (10 points)

(n+1)th polynomial will be as tight as nth degree. Because even if data set is small for (n+1)th degree, polyfit will disregard least coefficients.

Name ----------

7) Next to each item in the left column write the letter labeling the item in the right column that best matches the item in the left column. No item in the right column should be used more than once, and no box should contain more than one letter. (12 points)

[f]	data abstraction	a) inheritance
[b]	merge sort	b) divide and conquer
[a]	polymorphism	c) O(log n)
[e]	hashing	d) O(n)
		e) O(1)
		f) specification
		g) mutability

8) Do you think that the lectures are too slow paced, too fast paced, about right?

Too slow 1 2 3 4 5 Too fast

9) Do you think that the problem sets are too short, too long, about right?

Too short 1 2 3 4 5 Too long