



Assignment 10

Due Date for Programs: December 7, 2016

Problem 1 (10 points): What does this code do?

Take a look at the code below and the output that it produces. Try to figure out exactly what is going on. Explain what each line in the code does. Complete the worksheet at the end of this assignment. You should work on this with a partner.

```
1
2 import collections
3
4 numbers = [63, 54, 82, 59, 82, 5, 63, 63, 5, 66]
5
6 c = collections.Counter (numbers)
7
8 print ( c.most_common() )
9 print ( c.most_common(1) )
10 print ( c.most_common(2) )
11
12 n1 = c.most_common(1)
13 n2 = n1[0][0]
14 n3 = n1[0][1]
15
16 print (n2)
17 print (n3)
18
```

Output:

```
[(63, 3), (82, 2), (5, 2), (54, 1), (59, 1), (66, 1)]
[(63, 3)]
[(63, 3), (82, 2)]
63
3
```

Problem 2 (30 points): Stats on numbers

Write a program that prompts the user for a positive integer (this should be validated) and then generates a random list of values between 1 and 100 of that length. The program should print the information as shown in the samples below. You need to write three functions: **mean**, **median** and **mode**. Each of these functions should take a list as its argument and return the corresponding value. Note that the functions that calculate smallest and largest value in a list already exist in Python.

For calculation of mode you will need to use what you discovered in Problem 1.

Here are a few sample runs of the program:

Output:

```
Enter a number: 10
original list:
[53, 6, 30, 61, 19, 66, 38, 28, 74, 50]

mean is: 42.5
median is: 44.0
mode is: 50
smallest is: 6
```



```
largest is: 74
sorted list:
[6, 19, 28, 30, 38, 50, 53, 61, 66, 74]
```

Output:

```
Enter a number: 15
original list:
[89, 51, 93, 28, 76, 7, 73, 39, 45, 67, 42, 14, 49, 49, 57]

mean is: 51.93333333333333
median is: 49
mode is: 49
smallest is: 7
largest is: 93

sorted list:
[7, 14, 28, 39, 42, 45, 49, 49, 51, 57, 67, 73, 76, 89, 93]
```

Comment your source code by 1) briefly describing parts of your program 2) including your name, the date, and your class section at top of your file (above everything else) 3) **documenting all the functions following the IPO format**

What to Submit

This program should be named (i.e., the name of the file containing the program should be) `stats.py`. You only need to submit the source code for this problem.



Problem 3 (30 points): Counting Letters

Write a program somewhat similar to problem 3 in the previous lab. This program should count occurrences of all of the letters (not just vowels). You should NOT be keeping counters for every letter. Instead, you should work with two lists: one that contains the letters and one that contains the counts. Your `for` loops will need to iterate over indexes in these two arrays.

The program should prompt the user for the name of the file containing the book and then produce the results with counts of all of the letters as well as the sum of all the letters (this does not include the spaces, punctuation marks, digits or any other non-letter characters). Make sure that both upper and lowercase letters are counted.

Here is a sample run of the program:

Output:

```
Enter the file name: hamlet.txt

Total number of characters: 138860

Letter counts:
a: 10513    b: 1971    c: 2906    d: 5358
e: 16138    f: 2889    g: 2608    h: 8988
i: 9141     j: 150     k: 1315    l: 6195
m: 4487     n: 8864    o: 11985   p: 2252
q: 229      r: 8381    s: 8840    t: 12707
u: 4617     v: 1302    w: 3239    x: 270
y: 3437     z: 78
```

Output:

```
Enter the file name: alice.txt

Total number of characters: 123011
```



Letter counts:

a:	9804	b:	1746	c:	3003	d:	5469
e:	15396	f:	2383	g:	2944	h:	7890
i:	8634	j:	235	k:	1290	l:	5211
m:	2466	n:	8053	o:	9480	p:	1968
q:	220	r:	6612	s:	7269	t:	12204
u:	3979	v:	963	w:	2952	x:	176
y:	2584	z:	80				

Comment your source code by 1) briefly describing parts of your program 2) including your name, the date, and your class section at top of your file (above everything else) 3) **documenting all the functions following the IPO format**

What to Submit

This program should be named (i.e., the name of the file containing the program should be) `character_counter.py`. You only need to submit the source code for this problem.



Problem 4 (30 points): Prices and Taxes

Given the following list of prices, write a program that uses the list calculate two additional lists: one containing NYC sales tax and one containing combined price and the tax (remember that the sales tax is 8.875%).

```
prices = [29.99, 300.50, 5.15, 124.99, 325.00, 1000.34]
```

You should store the taxes and the prices with taxes in separate lists (so you need to have three lists in total) and display the results only after all the lists are calculated.

Here is a sample run of the program:

Output:

price	tax	price with tax
\$300.50	\$26.67	\$327.17
\$5.15	\$0.46	\$5.61
\$124.99	\$11.09	\$136.08
\$325.00	\$28.84	\$353.84
\$1,000.34	\$88.78	\$1,089.12

Comment your source code by 1) briefly describing parts of your program 2) including your name, the date, and your class section at top of your file (above everything else) 3) **documenting all the functions following the IPO format**

What to Submit

This program should be named (i.e., the name of the file containing the program should be) `prices.py`. You need to submit the source code as well as all the book text files for this problem.

Grading

The only way to receive the credit for the worksheet problems is to hand them in before the end of the lab session in which they are given.

The programs are graded based on correctness, style of code, design and documentation.

What and how to submit?

You should submit the source code file for each program to NYU Classes by the due date stated above. Make sure that you get an email confirmation after you submit the assignment. You should keep that email until the grades are returned - it is your proof that the assignment was submitted! If you do not get an email confirmation, you should try to resubmit the assignment. If you do not get that email, it means that we did not get your assignment.



Name(s) and NetId(s):

What does this code do?

line 2:

line 4:

line 6:

line 8:

line 9:

line 10:

line 12:

line 13:

line 14:

line 16:

line 17: