**CS 561 Python test**

1. Write a Python script (program) that will

Prompt the user for the radius of a sphere, after which the user will input a number r.

Output explanatory text (e.g. the words "Volume is") and the volume of the sphere.

According to my geometry book, the formula for the volume of a sphere is

(AWC the input is a positive #)

2 Write a Python script that will

Input positive ints until the user inputs a # that is divisible by 5

Output the sum of the even #'s input.

An example run of your program (with user input in **bold**) might go as

**47**

**3**

**8**

**4**

**20**

32

(AWC the user enters a sequence of positive ints terminated by the first multiple of 5; if you wish, you may assume that input has one # per line)

3. Write a Python script with the following functions:

f(x)

If x is a non-negative float, f returns 3 times the square root of x

without(s0, s1)

If s0 and s1 are strings, without returns a string that is the same as s0 except that it's missing all the chars in s1

For example, without('abracadabra', 'Yabba Dabba') would return 'rcdr

f1(li)

If li is a list of floats, f1 will (without changing li)

Create and return a new list that is the result of:

disregarding all values of li that are < 1

calling f on all values of li that are >= 1

assembling those values in a list in the opposite order of their corresponding values in the arg list.

For example, f1([0.5, -47.0, 9.0, 0.0, 4.0, 16.0]) will return [12.0, 6.0, 9.0]

f2(li)

If li is a list of nonempty lists of floats, f2 will (without changing li)

Create and return a new list parallel to li, where each element of the returned list is a tuple holding the min and max of the corresponding element of li

For example, f2([[1.0,2.0,3.0],[5.0],[10.0,9.0,8.0,7.0]]) will return [(1.0,3.0), (5.0,5.0), (7.0,10.0)]

4. Write a Python script that will

Prompt the user for an input file name (text mode)

Read the input file name. AWC the input is the name of a file in which every line consists of 1 or more positive ints separated by spaces

Prompt the user for an output file name (text mode)

Write one line to the output file for each line of the input file. The output line will be a sequence of ints and floats separated by spaces:

The line # (starting with 1)

All the #'s from the input file

The average of the #'s in the input file (which will be a float)

The median of the #'s in the input file (which will be a float)

For example, if the input file had the following 2 lines

10 13

20 5 7

then the output file would have something like the following 2 lines:

1 10 13 11.5 11.5

2 20 5 7 10.66666666667 7.0