Homework 2

Functions and Lists

Out 9/9 – Due 9/24

Exercise 1. Automate the boring stuff, page 77, The Collatz Sequence. 25pts.
Write a function named collatz() that has one argument called number. If number is even, then collatz() should print number // 2 and return this value. If number is odd, then collatz() should print and return 3 * number + 1. Then write a program that lets the user type in an integer and that keeps calling collatz() on that number until the function returns value 1. (Amazingly enough, this sequence actually works for any integer—sooner or later, using this sequence, you'll arrive at 1! Even mathematicians aren't sure why. Your program is exploring what's called the Collatz sequence, sometimes called "the simplest impossible math problem.") Remember to convert the return value from input() to an integer with the int() function; otherwise, it will be a string value. Hint: An integer number is even if number % 2 == 0, and it's odd if number % 2 == 1. The output of this program could look something like the following, and save your code as *hw2_firstname_lastname_ex_1.py:*

Enter number:
3
10
5
16
8
4
2
1

Exercise 2. Automate the boring stuff, page 77, Input Validation. 10pts.
Add try and except statements to the previous exercise (the collatz sequence) to detect whether the user types in a non-integer string. Normally, the int() function will raise a ValueError error if it is passed a non-integer string, as in int('puppy'). In the except clause, print a message to the user saying they must enter an integer. *Keep asking the user for an input until the input is an int*. Save your code as *hw2_firstname_lastname_ex_2.py*.

Exercise 3. List. 20pts.

1. Create a list with strings 'Wallet', 'Phone', and 'Keys', then print the list using a single line of code
2. Sort the list using the sort() function, then print the list again
3. Print the first item in the list
4. Print everything except the first item in the list (hint: slicing...) as a list
5. Print the last item in the list (hint: negative index)
6. Print the index of 'Keys' (hint: index() function)
7. Append 'Tablet' to the list, then print the list
8. Insert 'Mask' to the list as the second item in the list, then print the list
9. Remove 'Phone' from the list, then print the list
10. Reverse the list, then print the list
11. Finally, write a function strList() that takes this current list as an argument and returns a string with all the items separated by a comma and a space, with '*and*' inserted before the last item. For example, if the current list value is ['Tablet', 'Wallet', 'Mask', 'Keys'] (note this may not be the correct list at this point), then passing it to the function would return '*Tablet, Wallet, Mask, and Keys*'. Your function should be able to work with any list value passed to it.

12. Save your code as *hw2_firstname_lastname_ex_3.py*.

Exercise 4. Guess the number. 45pts (15pts each).

1. Develop a "Guess the number" game. Write a program that comes up with a random number between 1 and 20 (both inclusive), and the player has to guess it within 10 tries. If after 10 times the player's guesses were all wrong, print a message "Sorry, the number I was thinking of was xx". Otherwise, the program output should look like the following (save this code as *hw2_firstname_lastname_ex_4_1.py*):

I am thinking of a number between 1 and 20. You have 10 tries.
Take a guess.
10
Your guess is too low.
Take a guess.
15
Your guess is too low.
Take a guess.
17
Your guess is too high.
Take a guess.
16
Good job! You guessed my number in 4 guesses!

2. Complicate it by having random lower and upper bounds. Display a message "I am thinking of a number between lower and upper" and replace lower/upper with the random numbers you generated. The player still has to guess it within 10 tries. Make sure *your lower bound is not higher than you upper bound*. Save this code as *hw2_firstname_lastname_ex_4_2.py*.

3. Now, instead of user input, make the code guess it automatically. This automatic player still has to guess it within 10 tries. Make sure that for the automatic guesses, you don't use the same random number you already generated to guess. Save this code as *hw2_firstname_lastname_ex_4_3.py*.

Submit all your files **in a zipped archive named *hw2_firstname_lastname.zip***. Comment everything so we know it's your code. On top of each of your files please write this multiline comment:

'''
Homework 2, Exercise 1 (or 2, 3, …)
Name
Date
Description of your program.
'''