COMPSCI 590N: Assignment 1

Due: September 15, 2017 at 11:59pm

Please write all code for the following problems in the provided code file, `assignment1.py`, using the marked areas. Included with the assignment is a script for testing your solution called `assignment1_tests.py`. This script will test the output from your code against a number of test cases and will indicate if there are errors. Once you have written your code in `assignment1.py`, you can run these tests by executing:

```
python assignment1_tests.py
```

Be sure that you can run `assignment1_tests.py` in this way before submitting as this is how we will test your code for grading! The provided test cases are meant to help you debug your code, but you should not assume that they are exhaustive. If a problem asks you define a function or class, **you should use exactly the name specified** in the problem for this function or class. Your modified version of `assignment1.py` should be submitted to Moodle by the due date specified above.

Problem 1 (25 points)

Write the body for the function `evens_only` in `assignment1.py`. `evens_only` takes a list of numbers called `input_list` and returns a list of integers containing only the even, integer valued numbers in `input_list`. For example, the call `evens_only([1.0,2.0,2.5,3.0,4])` should return the list `[2,4]`. The numbers in the returned list should be in the same order as in `input_list`. You may assume that only lists of numbers will be passed as `input_list`, but you should make no assumptions about the type of the numbers (that is, they may be integers or floats). The returned value should be a list type and should contain integer types.

Problem 2 (25 points)

Write the body for the function `piecewise` that takes a single numeric input called `x` and implements the following piecewise defined function:

$$f(x) = \begin{cases} -1 & : x < 0 \\ 3x^2 & : 0 \leq x < 2 \\ -x & : 2 \leq x \end{cases}$$

You may assume the the input to `piecewise` will be a number, but the input may be either an integer or float. The output should have type float.

Problem 3 (50 points)

Write the body for the function `character_count` which takes as input a file name called `file_path`, reads in the text file at `file_path`, counts the number of times each alphabetic letter (a through z) is used in the input file, and returns a dictionary that maps each letter to the number of times it appears. Only letters used in the file should appear in the dictionary, punctuation and white space should be ignored, and letter case should be ignored. For example, if the file `test1.txt` contains:

```
Test
string!
```

then `character_count("test1.txt")` should return the dictionary:

```
{’e’: 1, ’g’: 1, ’i’: 1, ’n’: 1, ’s’: 2, ’r’: 1, ’t’: 3}
```

You may assume that `file_path` is a string and points an existing text file. The keys of the dictionary should have type `string` and the values should have type `int`. Three test files are included with the assignment for testing your code. Two useful resources for this problem are the built-in string methods (`https://docs.python.org/2/library/stdtypes.html#string-methods`) and the string module (`https://docs.python.org/2/library/string.html`).