FAYETTEVILLE STATE UNIVERSITY Department of Mathematics and Computer Science

CSC207 Assignment 4

In this assignment, you are going recode the `calcList` function in Python to solve the following problem using list comprehension.

Given a `float` value *v* and a list of `float` values *xs*, `calcList` returns a list of `float` according to the following rules:

1. For each value *x* in *xs*, if *x* is not positive, then will be no corresponding value in the output list.
2. Otherwise, the corresponding output value will be $x \cdot \ln x$.
3. However, this value will be in the output list if and only if its value is greater than *v*.
4. The order of the corresponding output values (if present) should be the same as the input values.

The followings are some examples:

- `calcList (1.0, [])` ➔ `[]`
- `calcList (1.0, [3.0])` ➔ `[3.2958]`
- `calcList (1.0, [-1.0, 1.0, 3.0, 5.0, 7.0, 9.0])` ➔ `[3.2958, 8.0472, 13.6214, 19.7750]`
- `calcList (100.0, range (1, 41))` ➔ `[102.0359, 106.4536, 110.9035, 115.3847, 119.8963, 124.4372, 129.0067, 133.6040, 138.2283, 142.8789, 147.5552]`

Download the three files "`csc207a4tester.py`", "`csc207a4testData.py`", and "`csc207a4.py`" from Canvas and save them in the same folder. The first two files are the tester program and the testing data file, respectively. Do not modify these two files. The last file is the file you will work on. Note that you cannot rename this file and you cannot change the name of the function (otherwise the tester will not be able to pick up your implementation). However, you can change the names of the parameters if you like.

The file "`CSC207a4.py`" contains a dummy implementation of the function that always returns the value of `None`.

The contents of the file "`CSC207a4.py`" is listed here for your reference:

```
from math import log

def calcList (v, xs):
    pass
```

The tester program will use the test data to test your implementation. It will run a total of 100 test cases. If your function is implemented correctly, running the tester should produce the following output:

```
Passed: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18,
19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36,
37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54,
55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72,
73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90,
91, 92, 93, 94, 95, 96, 97, 98, 99, 100]
Failed: []
100 of 100 test cases passed. Score = 20.00 of 20.00
```

If you would like, you can look up the test data from the file `csc207a4testData.py`. This file defines a single list named `tests`, which consists 100 tuples. Each tuple is a test, and it has 4 values inside:

- The first is an integer representing the test id.
- The second is the value of *v* representing the first input to the function.
- The third is a list representing the second input to the function.
- The last is the expected result of calling `calcList` using the input parameters.

The following is the first few lines in the test data file:

```
tests = [(  1, 18, [-83, 97, 37, 45, -16, …], [443.747, 133.604, 171.2998, …]),
         (  2, -99, [8, 58, -34, 39, -86, …], [16.6355, 235.5057, 142.8789, …]),
         (  3, -14, [29, -31, -19, -44, -67, …], [97.6516, …]),
         …
```

Notes:

1. Your program must be using the functional style; and using list comprehension. You will get a score of zero if you do not use list comprehension even your program can produce the correct result.
2. Your program will be tested with similar but different data.
3. You output should be correct up to 4 digits after the decimal point.
4. Your function ***CANNOT*** use recursion.
5. In general, you should not use variables. If you think you need to use variables, the variables should be assigned a value only once (this is to mimic functional programming).
6. You ***CANNOT*** define any other helper functions.
7. You can have only one `import` statement in your program; and that `import` statement has been provided. Any other `import` statement will cause you to lose points.

Submit only the file “`csc207a4.py`” to Canvas. The due date of this assignment is Sunday October 29.