CMPSC-132: Programming and Computation II
Spring 2019

Lab #7

Due Date: 02/22/2019, 11:59PM

Read the instructions carefully before starting the assignment. Make sure your code follows the stated guidelines to ensure full credit for your work.

Instructions:

- The work in this lab must be completed alone and must be your own.
- **Download the starter code file from the LAB7 Assignment on Canvas. Do not change the function names or given started code on your script.**
- A doctest is provided as an example of code functionality. Getting the same result as the doctest does not guarantee full credit. You are responsible for debugging and testing your code with enough data, you can share ideas and testing code during your recitation class.
- Each function must return the output (Do not use print in your final submission, otherwise your submissions will receive a -1 point deduction)
- Do not include test code outside any function in the upload. Printing unwanted or ill-formatted data to output will cause the test cases to fail. Remove all your testing code before uploading your file (You can also remove the doctest). Do not include the input() function in your submission.

Goal

[10 pts] In the starter code, there is a function called *triangle* that calls the function *recursive_triangle*(*n*,*n*) once. For this exercise you must write the recursive function *recursive_triangle*(*x*, *n*) that **returns** a string with the LAST *x* lines of a right triangle of base and height *n*.

- *recursive_triangle*(*x*, *n*) must be a recursive function, otherwise, no credit is given
- *recursive_triangle*(*x*, *n*) must return the pattern as ' ***\n   **\n   *'. As shown in the example, you can use the print method during testing to check if your pattern is correct.
- Don't modify anything in the triangle function. If your recursive function is correct, calling *triangle*(*n*) should return the complete right triangle.

```
>>> triangle(4)
'****\n ***\n  **\n   *'
>>> print(triangle(4))
****
 ***
  **
   *
```

This is the output that will be graded, make sure your function returns the value in this form

You can call the print method on your function to see the triangle pattern

```
>>> recursive_triangle(2,4)
'  **\n   *'
>>> print(recursive_triangle(2,4))
  **
   *
```

```
>>> triangle(6)
'******\n *****\n  ****\n   ***\n    **\n     *'
>>> print(triangle(6))
******
 *****
  ****
   ***
    **
     *
>>> recursive_triangle(3,6)
'   ***\n    **\n     *'

>>> print(recursive_triangle(3,6))
   ***
    **
     *
```

Notes:

- The doctest includes \\n instead of \n in order to keep consistent leading whitespace required by the shell session, your code must return output with as \n, otherwise it will fail the test cases!

Deliverables:

- Submit your code in a file name LAB7.py to the Lab7 GradeScope assignment before the due date