CSCE 110: Programming I

Lab #4 (100 points)

Due: Sunday, September 25th by 11:59pm

1 Please make sure you understand the following.

For this assignment, you are only allowed to use what we have discussed during the first 4 weeks of class.

Please label your Python programs `q<num>.py`, where `<num>` is the question number. Take your time and make sure you understand everything in this lab before getting started. Also, make sure your programs match the output EXACTLY as given for each question.

2 Lab Questions

1. *Drawing a rectangle.* Write a program (called q1.py) that asks the user for a symbol they would like to use to make a rectangle. Here, the symbol can be any single character: `&, #, +` , etc. Also, ask the user for the rectangle's width and height. Assume that the user enters a width and height greater than 2.

   Example #1. For the prompts of width, height, and symbol, the user enters 20, 3, and #, respectively (lines 1–3). The program then outputs the rectangle on lines 5–7.

```
1 Width: 20
2 Height: 3
3 Symbol: #
4
5 ####################
6 #                  #
7 ####################
```

Example #2. For the prompts of width, height, and symbol, the user enters 18, 8, and n, respectively (lines 1–3). The program then outputs the rectangle on lines 5–13.

```
1  Width: 17
2  Height: 8
3  Symbol: n
4
5  nnnnnnnnnnnnnnnnn
6  n                 n
7  n                 n
8  n                 n
9  n                 n
10 n                 n
11 n                 n
12 nnnnnnnnnnnnnnnnn
```

2. *Computing leap years.* Write a program (called q2.py) that asks for a starting year and an ending year and then prints all of the leap years between them (and including them, if they are also leap years). Leap years are divisible by 4. However, years divisible by 100 are not leap years unless they are also divisible by 400.

Example #1. The user enters 2000 and 2015 as the starting and ending years (lines 1–2). The program then outputs the leap years in this range (lines 4–8).

```
1  Starting year: 2000
2  Ending year: 2015
3
4  Leap years between 2000 and 2015
5  2000
6  2004
7  2008
8  2012
```

Example #2. The user enters 1881 and 1907 as the starting and ending years (lines 1–2). The program then outputs the leap years in this range (lines 4–9).

```
1  Starting year: 1881
2  Ending year: 1907
3
4  Leap years between 1881 and 1907
5  1884
6  1888
7  1892
8  1896
9  1904
```

3. *Drawing a rocket.* Write a program (called q3.py) to draw a rocket. The size of the rocket is an integer based on the user's input. Your program must work for a size of 3 or greater.

Example #1. The user is prompted for the size of the rocket ship, and the user enters 3 (line 1). Afterwards, the rocket ship is printed (lines 3–27).

```
1  Size: 3
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   /////**\\\\\
```

The size of the rocket ship refers to the subfigures in the middle of the output that have a height of 3. For example, lines 9–11 show a subfigure with a height of 3.

```
|../\..../\..|
|./\/\../\/\.|
|/\/\/\/\/\/\|
```

Lines 12–14, 16–18, and 19–21 are also subfigures of height 3. To draw the remaining parts of the rocket, the rest of the figure is scaled based on a size (height) of 3.

Example #2. Here, the user enters a size of 5 at the prompt (line 1). The output of the rocket is shown in lines 3–43.

```
1  Size: 5
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    /////////**\\\\\\\\\
```

Lines 13–17 show a subfigure with a size or height of 5.

```
|..../\........./\....|
|.../\/\......./\/\...|
|../\/\/\..../\/\/\..|
|./\/\/\/\../\/\/\/\.|
|/\/\/\/\/\/\/\/\/\/\|
```

Lines 18–22, 24–28, and 29–33 are also subfigures of height 5. Again, the rest of the figure is scaled based on a size of 5.

Programming tips. Do not attempt to write this program all at once! It is best completed in stages. That is, use divide and conquer strategy. Here is a recommended strategy.

a) Examine the output and figure out the patterns of repeated characters on each line. Do this before writing any code.

b) Write the program to draw a rocket of size 3. Use while loops to produce the patterns in the output. Also, test your program frequently as you write your code in a step-by-step fashion.

c) Extend your program from part (b) to draw a rocket of any size greater than or equal to 3.

Example #3. Just in case you needed a larger example. Here is a big rocket of size 14!

```
Size: 14

                                /**\
                               //**\\
                              ///**\\\
                             ////**\\\\
                            /////**\\\\\
                           //////**\\\\\\
                          ///////**\\\\\\\
                         ////////**\\\\\\\\
                        /////////**\\\\\\\\\
                       //////////**\\\\\\\\\\
                      ///////////**\\\\\\\\\\\
                     ////////////**\\\\\\\\\\\\
                    /////////////**\\\\\\\\\\\\\
                   //////////////**\\\\\\\\\\\\\\
                  ///////////////**\\\\\\\\\\\\\\\
                 ////////////////**\\\\\\\\\\\\\\\\
                /////////////////**\\\\\\\\\\\\\\\\\
               //////////////////**\\\\\\\\\\\\\\\\\\
              ///////////////////**\\\\\\\\\\\\\\\\\\\
             ////////////////////**\\\\\\\\\\\\\\\\\\\\
            /////////////////////**\\\\\\\\\\\\\\\\\\\\\
           //////////////////////**\\\\\\\\\\\\\\\\\\\\\\
          ///////////////////////**\\\\\\\\\\\\\\\\\\\\\\\
```