Assignment 2 - Pyramids

Start Assignment

- Due Saturday by 11:59pm
- Points 20
- Submitting a file upload
- File Types java
- Available Jan 5 at 12am - Jan 26 at 11:59pm

2 Assignment 2 - Pyramids

Computer SCIENCE

IMPORTANT: Please review the statement on Academic Integrity. Submitting code, without permission, that is not your own may result in an automatic F for the class: Statement on Academic Integrity (https://usu.instructure.com/courses/797539/pages/statement-on-academic-integrity)

In this assignment you will write two different programs, each generating a pyramid of numbers. This will have you create a basic Java program, accept and user user input, convert from numbers to strings, write nested loops, and generate formatted output.

Write two programs that produce the number pyramids as shown below.

The following video provides some key insights for writing these two programs...

Pyramid 1

Name this program **Pyramid1.java**. Two examples from this program...

```
Enter the number of lines: 8
               1
             2 1 2
           3 2 1 2 3
         4 3 2 1 2 3 4
       5 4 3 2 1 2 3 4 5
     6 5 4 3 2 1 2 3 4 5 6
   7 6 5 4 3 2 1 2 3 4 5 6 7
 8 7 6 5 4 3 2 1 2 3 4 5 6 7 8
```

```
Enter the number of lines: 12
                                 1
                              2  1  2
                           3  2  1  2  3
                        4  3  2  1  2  3  4
                     5  4  3  2  1  2  3  4  5
                  6  5  4  3  2  1  2  3  4  5  6
               7  6  5  4  3  2  1  2  3  4  5  6  7
            8  7  6  5  4  3  2  1  2  3  4  5  6  7  8
         9  8  7  6  5  4  3  2  1  2  3  4  5  6  7  8  9
      10  9  8  7  6  5  4  3  2  1  2  3  4  5  6  7  8  9 10
   11 10  9  8  7  6  5  4  3  2  1  2  3  4  5  6  7  8  9 10 11
12 11 10  9  8  7  6  5  4  3  2  1  2  3  4  5  6  7  8  9 10 11 12
```

Each column of numbers takes up the same space, based on the largest value in the pyramid. The first example, the largest value takes up 1 space, therefore all columns are 2 spaces. The second example, the largest number is 12, which takes up 2 spaces, therefore all columns are 3 spaces. At

first the second pyramid (in the above examples) doesn't quite look right in the transition from 10 to 9 and when comparing the left and right sides, but it is what it being requested for the assignment.

Pyramid 2

Name this program **Pyramid2.java**. Two examples from this program...

```
Enter the number of lines: 8
                          1
                      1   2   1
                  1   2   4   2   1
              1   2   4   8   4   2   1
          1   2   4   8  16   8   4   2   1
      1   2   4   8  16  32  16   8   4   2   1
  1   2   4   8  16  32  64  32  16   8   4   2   1
```

```
Enter the number of lines: 12
                                                  1
                                              1   2   1
                                          1   2   4   2   1
                                      1   2   4   8   4   2   1
                                  1   2   4   8  16   8   4   2   1
                              1   2   4   8  16  32  16   8   4   2   1
                          1   2   4   8  16  32  64  32  16   8   4   2   1
                      1   2   4   8  16  32  64 128  64  32  16   8   4   2   1
                  1   2   4   8  16  32  64 128 256 128  64  32  16   8   4   2   1
              1   2   4   8  16  32  64 128 256 512 256 128  64  32  16   8   4   2   1
          1   2   4   8  16  32  64 128 256 512 1024 512 256 128  64  32  16   8   4   2   1
      1   2   4   8  16  32  64 128 256 512 1024 2048 1024 512 256 128  64  32  16   8   4   2   1
```

Additional Information

- The programs must prompt the user for how many lines to generate. Your program must be able to generate properly formatted output regardless of the number of lines requested. Notice the formatting of the output is based upon the maximum number generated in each of the pyramids. The number generated will not be larger than a `long`; realistically probably not more than 30 (rows).
- You can convert a number to a string like: `String str = "" + value;` Use this technique to convert the user input to a string, and then get the number of characters in the string to determine the formatted output field width
- The format specifier for formatted output (`System.out.printf`) can be a `String` variable. You'll need to do this to make sure the output is formatted based on the size of the numbers.

Notes & Submission

- Turn in only the source code files named **Pyramid1.java** and **Pyramid2.java**

- You may assume the user provides valid input. You do not need to check for errors in the input such as letters, or crazy numbers of rows.
- Your code must compile with no errors or warnings. See syllabus regarding submitting code that has compile errors.
- Your code must adhere to the CS 1410 coding standard: **link (https://usu.instructure.com/courses/797539/pages/cs-1410-coding-standard)**
- Java SDK Docs: **link (https://docs.oracle.com/en/java/javase/25/index.html)**

Assignment 2 - Rubric

Criteria	Ratings		Pts
Compiles without any warnings	**2 pts** **Full Marks**	**0 pts** **No Marks**	2 pts
Follows required course code style	**2 pts** **Full Marks**	**0 pts** **No Marks**	2 pts
User input	**2 pts** **Full Marks**	**0 pts** **No Marks**	2 pts
Overall Pyramid 1 display Display of the first pyramid, even if the formatting isn't perfect.	**3 pts** **Full Marks**	**0 pts** **No Marks**	3 pts
Overall Pyramid 2 display Display of the second pyramid, even if the formatting isn't perfect.	**3 pts** **Full Marks**	**0 pts** **No Marks**	3 pts
Perfect Pyramid 1 display Ability to handle any number of lines	**4 pts** **Full Marks**	**0 pts** **No Marks**	4 pts
Perfect Pyramid 2 display Ability to handle any number of lines	**4 pts** **Full Marks**	**0 pts** **No Marks**	4 pts

Total Points: 20