Important: In this lab, you will begin writing loops. You will inevitably write some infinite loops that will cause your BlueJ environment to freeze. When this happens, hold down the `ctrl` key and press `d`. *Switch partners between every problem!*

**Part I: Loop Exercises**

- Create a New Project named `Lab4`. For each of the following problems, **create a new class**.
- Write a program that inputs a positive integer $n$, then prints 1, 2, ..., $n$ each on a separate line.
- Write a similar program as the one above, but prints $n$, $n-1$, ..., 1 each on a separate line.
- Write a program `Odds` that asks a user for any positive integer, $k$ and $n$. Starting with $k$, your program should print the next $n$ odd numbers. You may assume $k > 0$ and $n > 0$, but you may not assume that $k$ is odd or even. For instance, $k = 4, n = 4$ should output 5, 7, 9, 11, and $k = 31, n = 3$ should output 31, 33, 35.
- Write a program `SimpleStats` that first inputs a positive integer $n$. This time, **you may not** assume $n > 0$, so continue asking the user for $n$ until they enter a positive number. After you’ve ensured that the user enters a positive number for $n$, you need to then obtain $n$ integers from the user. Afterwards, output the maximum value entered, minimum value entered, and the average value.

Enter a value: -4
This value must be positive!

Enter a positive value: 0
This value must be positive!

Enter a positive value: 3

-- Thank you! The program will now input 3 values. --

Enter value 1: 4
Enter value 2: -4
Enter value 3: 9

The maximum value entered is: 9
The minimum value entered is: -4
The average is: 3
Part II: “I’m Thinking of a Number between...”

☐ This program takes a little more time to design and implement. Switch roles every 15-20 minutes.

☐ Write a program GuessingGame that runs the popular childhood game, I’m thinking of a number...

☐ Your program will select a random integer in the range [1, 100]. We’ll call this integer answer. Hint: First use Math.random() to generate a number in [0, 1).

☐ The program will then proceed ask the player for guesses. The program will not terminate until the answer has been guessed correctly.

☐ If the guess is wrong, your program must let the user the know if it was too low, or too high.

☐ When the user guesses correctly, output the number of tries and one of the following messages:

- 1 guess: “Hey, that was lucky!”
- 2-4 guesses: “That was amazing!”
- 5-6 guesses: “That was pretty good.”
- 7 guesses: “Eh... you could do better.”
- 8-9 guesses: “That was pretty bad!”
- 10 or more: “Wow, you’re really terrible at this.”

I’m thinking of a number between 1 and 100.
What’s your guess? 25
Lower!
What’s your guess? 10
Lower!
What’s your guess? 4
Higher!
What’s your guess? 6
Higher!
What’s your guess? 7
Higher!
What’s your guess? 8

You got it in 6 tries. That was pretty good.

Part III: Submitting Your Lab

After you have completed the lab, please do the following to submit your work.

☐ Zip up all necessary files.

☐ Rename the zip file to LastName1_LastName2_Lab4.zip, where the LastNames are the respective last names of the authors.

☐ Submit it on Moodle. You may submit as often as you’d like before the deadline. I will grade the most recent copy.