Physics World – Part II

An experimental 2-dimensional, physics-heavy, sandbox-game-thing.

# Submission Guidelines:

In GitHub, create a branch for your project named “Part2\_Completed”.

Make no further changes to this branch. Make no further branches from this branch. This branch is frozen – a monument to the first step on this project. I will clone that branch, and create a new repo, at the time the assignment is due.

Due at the start of our next class meeting

# Base Requirements (83)

* In GitHub, create a branch for your project named “Part2\_Completed”.
* Place an annotated version of this rubric in the root directory of your Unity project. Save it in Word .docx format. When I download your Part1\_Completed branch, I should find the rubric.
* Build a LoopNumberer script that runs on Start and does the following things:
  + Prints the odds from 1 to 99, all with one Debug.Log: 1, 3, 5, .... 99
  + Prints the fives from 55 to 255, all with one Debug.Log: 55, 60, 65, 70, ... , 255
  + Prints the evens from -100 to 0, all with one Debug.Log: -100, -98, -96, ... -2, 0
* Build the PatternMaker\_UniformRandom component, and use it in your project somewhere:
  + Explain: Where have you used this component in your project?
* Build the PatternMaker\_StraightLine component, and use it in your project somewhere:
  + Explain: Where have you used this component in your project?

# Stretch Goals:

* Upgrade your LoopNumberer to create some more complex sequences:
  + (+8) Print the 10’s from 10 to 100, alternating +/-: 0, -10, 20, -30, 40, -50, etc.

I get the rules：-1^n-1 \* 10(n-1)

* + (+10) Print every number between 0 and 1000 that is divisible by either 6 or 13:

6, 12, 13, 18, 24, 26, ... etc. hint:

The % operator, called ‘modulus’, will probably be helpful

My rules: 13n or 16n?

* + (+ 15) Prints the prime numbers from 0 to 500: 1, 3, 5, 7, 13, 17, ... etc

!!! Your script must figure out whether a number is prime or not. No hard-coding the primes.

* (+20) Build a PatternMaker\_Piecewise component, based on the StraightLine component.
  + The component should use ifs to make a piecewise line with at least \*three\* pieces ([Wikipedia](https://en.wikipedia.org/wiki/Piecewise))
  + You should use the component somewhere in your project
  + Explain: What are the pieces of your line?
  + Explain: Where have you used this component in your project?
* (+10) Build a PatternMaker\_RandomWalk component, based on the Spiral component
  + The component should draw a random line – take a step, turn by a random amount, take another step, turn by a random amount, repeat ... ([Wikipedia](https://en.wikipedia.org/wiki/Random_walk))
  + Explain: What does your comInteresting usage of different joint-type components. Outline the usage below: What joint-component did you use, and what did you build with it?
* (+5 to +20) Build some other sort of PatternMaker component, and use it in your project.
  + You are welcome to use my code as a starting point
  + AND then your code should do something new. It should build on the code that I have shared.
  + Explain: What pattern does your script make?
  + Explain: Where have you used this component in your project?
* (+10 to +15 pts) Dig into SpriteShapes and SpriteShapeControllers.
  + Build a SpriteShape that uses a Controller with multiple sprites
  + The controller should have at least one fill sprite, and at least 2 edge sprites
  + Explain: Where have you used this object in your project?
* (+1 to +20) Other. Something nifty related to number sequences, spawn patterns, for loops, or SpriteShapes.
  + Explain: What is your nifty thing?
  + Explain: Where have you used this thing in your project?