CS106Ax Assignment 1

Esteban Rey L
In order to run and debug the code you will be using a web browser. We strongly recommend you use **chrome** for this as it has the best developer tools.
1 - Luhn's Algorithm

Start from the right

Extract each digit

For every second digit from the right double it. If the number is doubled and exceeds 9 subtract 9.
<table>
<thead>
<tr>
<th>Input</th>
<th>Luhn Digit Sum</th>
</tr>
</thead>
<tbody>
<tr>
<td>596825</td>
<td>5</td>
</tr>
<tr>
<td>596825</td>
<td>2*2 = 4</td>
</tr>
<tr>
<td>596825</td>
<td>8</td>
</tr>
<tr>
<td>596825</td>
<td>6*2 = 12 - 9 = 3</td>
</tr>
<tr>
<td>596825</td>
<td>9</td>
</tr>
<tr>
<td>596825</td>
<td>5*2 = 10 - 9 = 1</td>
</tr>
</tbody>
</table>
5 + 4 + 8 + 3 + 9 + 1 = 30

If the number ends in 0 then this is a valid credit card number
Tips and Tricks

Pay attention to the % operator. This operator returns a remainder. For example if you perform 10559 % 10 this will give you a 9.

You may also find some functions of the math library to be useful.

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Math.PI</td>
<td>The mathematical constant π</td>
</tr>
<tr>
<td>Math.E</td>
<td>The mathematical constant e</td>
</tr>
<tr>
<td>Math.abs(x)</td>
<td>The absolute value of x</td>
</tr>
<tr>
<td>Math.max(x, y, ...)</td>
<td>The largest of the arguments</td>
</tr>
<tr>
<td>Math.min(x, y, ...)</td>
<td>The smallest of the arguments</td>
</tr>
<tr>
<td>Math.round(x)</td>
<td>The closest integer to x</td>
</tr>
<tr>
<td>Math.floor(x)</td>
<td>The largest integer not exceeding x</td>
</tr>
<tr>
<td>Math.log(x)</td>
<td>The natural logarithm of x</td>
</tr>
<tr>
<td>Math.exp(x)</td>
<td>The inverse logarithm (e^x)</td>
</tr>
<tr>
<td>Math.pow(x, y)</td>
<td>The value x raised to the y power (x^y)</td>
</tr>
<tr>
<td>Math.sin(θ)</td>
<td>The sine of θ, measured in radians</td>
</tr>
<tr>
<td>Math.cos(θ)</td>
<td>The cosine of θ, measured in radians</td>
</tr>
<tr>
<td>Math.sqrt(x)</td>
<td>The square root of x</td>
</tr>
<tr>
<td>Math.random()</td>
<td>A random value between 0 and 1</td>
</tr>
</tbody>
</table>
2 - Hailstone

1. Pick some positive integer and call it \( n \).
2. If \( n \) is even, divide it by two.
3. If \( n \) is odd, multiply it by three and add one.

Continue this process until \( n \) is equal to one.
15 is odd, so I make $3n+1$: 46
46 is even, so I take half: 23
23 is odd, so I make $3n+1$: 70
70 is even, so I take half: 35
35 is odd, so I make $3n+1$: 106
106 is even, so I take half: 53
53 is odd, so I make $3n+1$: 160
160 is even, so I take half: 80
80 is even, so I take half: 40
40 is even, so I take half: 20
20 is even, so I take half: 10
10 is even, so I take half: 5
5 is odd, so I make $3n+1$: 16
16 is even, so I take half: 8
8 is even, so I take half: 4
4 is even, so I take half: 2
2 is even, so I take half: 1
3 - Stern Brocot Sequences

This one is more algorithmically challenging
- Our goal is to find the sequence that leads to a particular fraction starting from $\frac{1}{2}$

- This results in R L L L L L R R L type of sequences

- Every new branch is built by either taking a right or a left.
  - A left takes you to a smaller fraction
  - A right takes you to a larger fraction
Tips and Tricks

Note that you do not have and in fact should avoid generating the whole tree.

There are 3 items that must be kept track of before generating every level, the Top Left Fraction, the Top Right Fraction and the Current Fraction.

Avoid storing the fractions as javascript decimal numbers this can be more confusing.
Graphics Library

See documentation at https://jdkula.dev/jsgraphics-docs/

JSGraphics

The JSGraphics library can be used to easily manipulate an HTML5 Canvas, and is used by Stanford's CS106AJ curriculum.

For students: How to read this documentation

This document is intended to read like documentation you might find for any library online. It's a good thing to be able to read it, and see what such documentation might look like. If you're ever creating much larger projects, you're very likely to be reading documentation just like this!

Because this is likely the first time you're looking at documentation like this, you'll see some notation and vocabulary you haven't seen before. I'll briefly go over this so you this documentation will be useful to you!

Vocabulary

Some of these definitions may seem scary at first. Don't worry, I'll give an example below.

<table>
<thead>
<tr>
<th>Word</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class</td>
<td>A blueprint for an object. A class says what an object will contain, and also defines all of its functionality. If an object matches a class, we say that object is an instance of that class.</td>
</tr>
</tbody>
</table>
Canvas Coordinate System

- 0,0 is the top left corner
- Y increments downwards
- X increments to the right
- Negative values are valid and will simply place part or all of a G-Object outside of view
- All G-Objects are place with respect to their top left corner
- Order matters, items added earlier will be behind items added later if they overlap.
GCompound Coordinate System

- Not Required but probably useful
- More akin to a traditional coordinate plane
- Probably want to use it for problem 5
4 - Rainbow

Make a Rainbow

GRect

GOvals
5 - Sampler Quilt

4 Shapes which repeat and loop

Bullseye
Log Cabin
Random Colored Flowers
Random Section Leader
7x7 quilt
Tips and Tricks

Make sure to decompose! You definitely want functions that will generate each pattern.

GCompounds will make your life easier when making these values

Loops can be nested, that is:

```javascript
For (let i ....) {
  For (let j ...)
}
```

Use loops for style and ease!
Advanced Debugging Techniques with VS code
Setting up launch.json

```
{
  // Use IntelliSense to learn about possible attributes.
  // Hover to view descriptions of existing attributes.
  // For more information, visit: https://go.microsoft.com/fwlink/?linkid=830387
  "version": "0.2.0",
  "configurations": [
    {
      "type": "chrome",
      "request": "launch",
      "name": "Launch Chrome",
      "file": "${file}",
      "webRoot": "${workspaceFolder}" 
    }
  ]
}
```
Run with f5!