Peer Review 2

Usability of code

* Appropriate parameterisation including defaults
* Encapsulation (private fields where appropriate)
* Useful methods including draw

The code has appropriate parameters for the Spirograph class. It has useful methods including getters and setters outside of the draw and setup function to set values. They have an additional updateloopc function as well.

Within the Spirograph.js file, the developer has appropriate default variables but has included this.updateloopc() when it is not required when there is a function already defined with the same name. Within index.js the developer has getvalue() which gets the values from the HTML document and calls the setters within Spirograph.js class.

The index.js file has some global variables such as R, r and rho which are then passed into the new object when they could be defined only within the spirograph class instead as there are setters and getters for such variables.

**Development of original**

* Original code to be submitted as well as final version
* Work done in refactoring code to class
* Work done in useful parameterisation
* Work done in extending scope

The developer has included the original code with the final version.

Code was already in JavaScript, however the person has created a separate class from index.js when refactoring it. The spirograph has its own parameters within its constructor. There is some useful parameterisation, for example having individual setters for each HTML form and passing ‘R’, ‘r’ and ‘rho’ into the Spirograph constructor. The code also includes the optional p5 renderer which is checked for in the draw function when g (the renderer) is passed as a parameter.

The user has extended the code to be able to change the size of the circles and the position of where it is drawn. The brush size can also be changed.

**Quality of example**

* HTML page is valid
* Appropriate on-page instructions
* Appropriate on-page controls (form)

HTML page is mostly valid except for one error where the character encoding was not declared.

The example included in the HTML is working apart from “size of moving circle”. I am assuming it was supposed to have the same effect as the original code where the shape would keep changing. The inputs are within the HTML file do still all work and affect the shape.

However, there is no instructions or examples to help the user with what each input does and how to interact with the webpage. There are general titles but not do not include a range or helpful suggestions.

**Quality of documentation**

* All methods and parameters explained (including constructor)
* Explanation of example
* Source of initial code acknowledged (including licence)

Documentation goes through and explains all parameters within the Spirograph constructor. They explain in short detail most of what each method does but not all the parameters that are passed in. There are some methods such as ‘lcm’ and ‘gcd’ which are not mentioned in the documentation or commented within the code.

It does not contain any link or reference to either the original code or the licence as well. The explanation of the example is done through a picture with a short description saying what happens with the input forms. There is not much detail as to what each input does or how it works apart from saying the draw function is the result of what is shown.

**Code quality: ESLint**

Apply rules from [eslint.org/docs/rules/](https://eslint.org/docs/rules/):

* Possible Errors
* Best Practices
* Variables
* Stylistic Issues
* ECMAScript 6

The code has missing semicolons e.g. line 3,9,14 in spirograph.js. Another error is that the person has used tabs instead of 4 spaces when indenting and has used double quotes instead of single quotes. When commenting, there are some that are in line with the code itself. Apart from the stylistic issues, there is nothing else wrong with the code quality.