1. Let's create some aRt!

   (a) Install the `aRtsy` package. Provide the code in an R chunk that does not run. You only need to install it one time.
   Solution:

   ```
   install.packages("aRtsy")
   ```

   (b) Load the `aRtsy` package. Provide the code in an R chunk that does run. We need to load the library each time it is run.
   Solution:

   ```
   library(aRtsy)
   ```

   (c) Running `demo("aRtsy")` or `vignette("aRtsy")` don't return any helpful demos or tutorials. However, if you run `help("aRtsy")` you will find a link to a tutorial. Recreate the first figure they make using `canvas_collatz()`. Make sure to update the caption.
   Solution:

   ```
   help("aRtsy")
   set.seed(1)
   canvas_collatz(colors=colorPalette("tuscany3"))
   ```

Figure 1: A collatz conjecture with a seed of 1

   (d) Change the randomization seed to 1313, which will change the random numbers generated to create the plot. Can you see the difference? Make sure to update the caption.
   Solution:

   ```
   set.seed(1313)
   canvas_collatz(colors=colorPalette("tuscany3"))
   ```

Figure 2: A collatz conjecture with a seed of 1313

(e) Now, create a new Collatz conjecture plot by specifying the following arguments. Note you will find the help file for the `canvas_collatz()` function to be rather helpful. Make sure to update the caption.

- Use the `vrolik4` color palette. Note you can find other by running `?colorPalette` in the console.
- Make the background grey. Note a hexcode for grey is `#dbdbdb`.
- Specify that there should be 72 strands.
- Specify the angle used for bending the sequence for odd numbers as -0.05.
- Specify the angle used for bending the sequence for even numbers as 0.0145 (note this is the default).

Solution:

```
canvas_collatz(colors=colorPalette("vrolik4"),
               background = "#dbdbdb",
               n = 72,
               angle.even = 0.0145,
               angle.odd = -0.05)
```

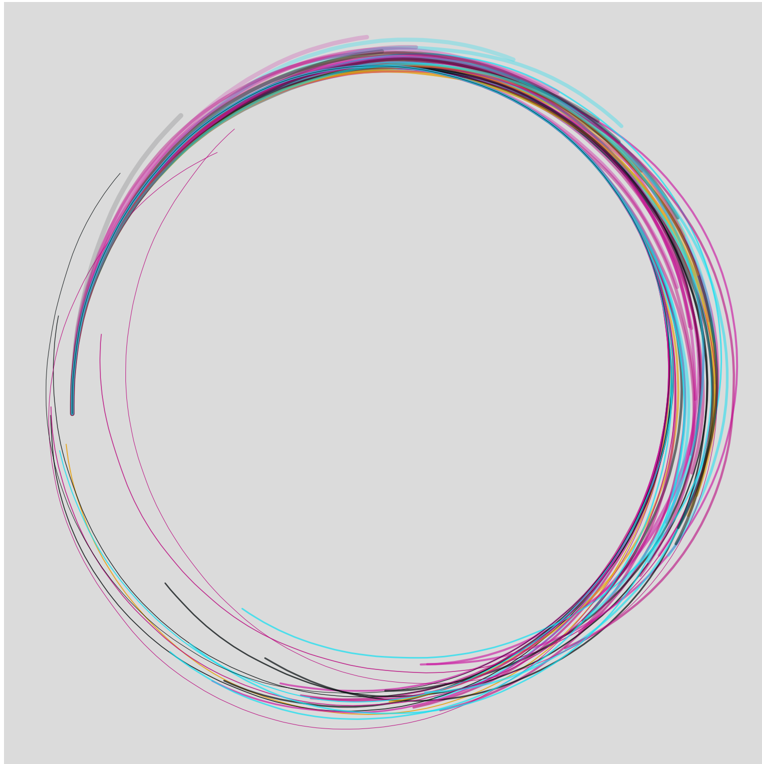

Figure 3: A collatz conjecture with a seed of 1313, 72 strands, and -0.05 angle for bending the odd number sequence.

(f) Make another plot using the tutorial – feel free to be creative here! Note that I leave creating the R chunk and figure environment to you here. Make sure that your code is well-formatted and your plot is appropriately scaled.
Solution:

```
set.seed(149)
colors <- list(
  c("darkseagreen", "darkslategray4","deepskyblue4","azure"),
  c("sienna3","peru",’lightsalmon1’,"ghostwhite"),
  c("orangered2","orange", "tomato1","coral"),
  c("ivory","azure","mintcream","grey","grey100","lightgrey")
)

canvas_planet(colors,
              starprob=0.005,
              radius = c(300, 200, 1500, 100),
              center.x= c(750, 1900, -1100, 1200),
              center.y= c(700, 700, 1000, 700),
              light.right = FALSE,
              resolution = 2000
              )
```

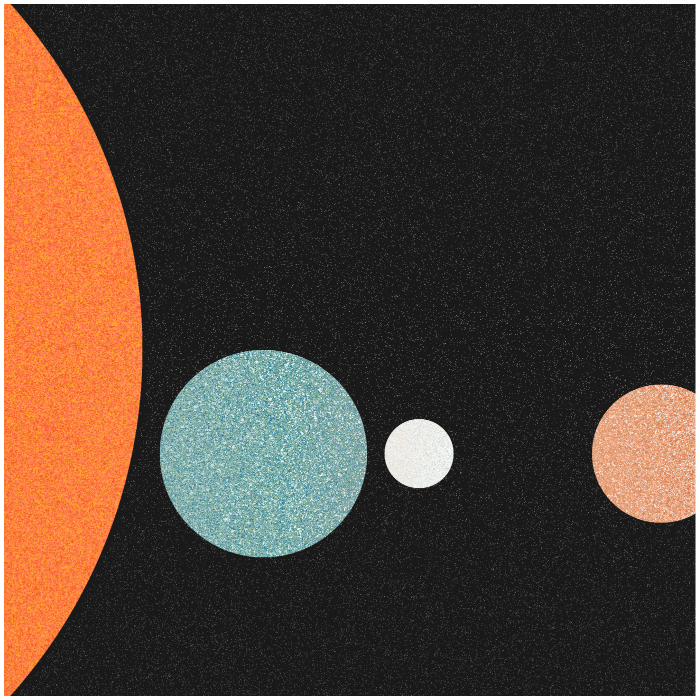

Figure 4: A planet plot of an Earth-like planet, a moon, and a Mars-like planet.

(g) Use `citation()` to get the BiBTeX citation for the `aRtsy` package and use `\citep{}` to add a parenthetical citation to the end of the sentence below. **Solution:** We created the generative art in Question 1 using the `aRtsy` package for `R` (**?**).

2. Suppose we wanted to solve $2^{x+1} + 2^{x-1} = 40$ for x. While this is a pretty straightforward algebra problem, it's useful for demonstrating the use of objects in R.

   (a) Create a numeric vector containing the integers from 0 to 10 inclusive. Hint – the solution to this problem is one of these values.
   Solution:

   ```
   vector1 = c(0:10) #creates a vector of the integers from 0 to 10
   ```

   (b) Complete the algebra to compute $2^{x+1} + 2^{x-1}$ for each value in the numerical vector created in step 1. Make sure to save the result to a new numeric vector.
   Solution:

   ```
   completed_algebra <- c()
   for (x in vector1){
     equation1 <- 2^(x+1) + 2^(x-1) #left side of the equation; = 40
     completed_algebra <- append(completed_algebra, equation1)
   }
   ```

   (c) Use the which() function to ask which result is 40.
   Solution:

   ```
   answerQ2 <- which(completed_algebra == 40)
   ```

   (d) What is the solution? That is, what value of x yields $2^{x+1} + 2^{x-1} = 40$?
   Solution:

   $x = 4$

   (e) Explain why this approach wouldn't work for something like $3^{x+2} + 5(3^x) = 84$ where the solution is $x \approx 1.6309$.
   Solution: This approach assumed that x is an integer between 0 and 10. We began and based all of our work off of that when we created the vector from 0-10 and tried every value in it, to see which was correct. This is more akin to "guess and check" than actual algebra, which is needed in more complicated problems.