A mental exercise for when you’re stuck doing something menial

If there’s one thing I learned from teaching, it’s that one of the most mind numbing activities is test proctoring. Most proctors aren’t allowed to use their smartphone, read a book, draw, or even sit. You are supposed to walk around and “actively proctor,” whatever that means. To cope, one of the tricks I learned was to attempt to determine the collatz sequence of a random starting number, mentally.

A collatz sequence is a simple algorithm that creates a chain of numbers based on a starting number. The rule is simple:

If n is even then:

n 🡪 n/2

If n is odd then:

n 🡪 3n + 1

For example if you start with 19 you would get

[show chain here]

This is called the collatz *conjecture* because it’s unproven whether or not any starting number will always reach one, like in the example above. Here is the question from project Euler [link] which I will try to solve using Ruby.

What starting number produces the longest chain?

Algorithm pseudocode:

Write a helper function that will determine the next number in the sequence

Write a function that will loop through every number keeping track of the number of iterations.

Once the chain has finished, check if it has surpassed the last winner. If it has, then replace it as the current ‘leader’ .

Let’s try to solve this in Ruby