Homework #1

High Performance Scientific Computing

AMath 483/583

The first homework assignment will be written entirely in Python. It will serve, in part, as practice for submitting homework as well as a test of your Python, Numpy, and Matplotlib knowledge and your general programming abilities.

For the portions of the problems marked "**Report**" be sure to include your written responses and any plots requested in a PDF document. Place the document in the "`report`" directory of your repository and give it the name "`report.pdf`". Please include your name and UWNetID at the top of the report.

Exercise #1

Consider the function $C : \mathbb{N} \to \mathbb{N}$

$$C(n) = \begin{cases} n/2, & \text{if } n \equiv 0 \pmod 2, \\ 3n+1, & \text{if } n \equiv 1 \pmod 2, \\ 1, & \text{if } n = 1. \end{cases}$$

That is, if the input number is even then divide it by two. If the if the input number is odd then multiply it by three and add one. If the given number is one then return one.

Given an $n \in \mathbb{N}$ define the sequence,

$$n_0 := n, \;\; n_1 := C(n_0), \;\; n_2 := C(n_1), \;\; n_3 := C(n_2), \;\; \ldots.$$

The *Collatz conjecture* states that no matter which $n \in \mathbb{N}$ you begin with the sequence of integers n_k will be finite and end with the number one. That is,

$$S(n) := \{n_0, n_1, \ldots, n_m = 1\}.$$

This sequence is called a *Collatz sequence* and the number m is called its *stopping time.*

For example, the Collatz sequence of $n = 6$ is equal to,

$$\S(6) = \{6, 3, 10, 5, 16, 8, 4, 2, 1\},$$

and therefore its stopping time is equal to $m = 8$.

1. Write the body of the function `collatz_step(n)` defined in the module `homework1.exercise1.collatz_step`. (That is, in the file `homework1/exercise1.py` located in the homework repository.) The value of this function should be equal to the function $C(n)$ defined above.

2. Write the body of the function `collatz(n)` corresponding to the funciton $S(n)$ defined above. This function should output the Collatz sequence corresponding to the input $n > 0$.

3. **Report:** For your homework report, create a scatter plot of stopping times for all $1 \leq n \leq 5000$. That is, the input integer n should be on the horizontal axis and the stopping time should be on the vertical axis. *Be sure to include appropriate labels and a title. Make the color of the data points blue.* Do you think based on this picture that the Collatz conjecture is true? Say something about how the picture might be sufficient or insufficient to make the conjecture.

4. **Automated Tests:** The hidden test suite will test your implementations of `collatz_step` and `collatz` in the following way:

    - Does `collatz_step` produce the correct output on a variety of examples?
    - Does `collatz_step` handle the case when $n = 1$ properly?
    - Does `collatz` produce the correct output on a variety of examples?

Exercise #2

Many scientific computation problems involve finding the root of a given function. Newton's method is a simple, yet highly effective and popular way to numerically approximate a "nearby" root x^* of a function $f : \mathbb{C} \to \mathbb{C}$. Define the *Newton step function*,

$$N(f, x_0) = \begin{cases} x_0 - \frac{f(x_0)}{f'(x_0)}, & \text{if } f'(x_0) \neq 0, \\ x_0, & \text{if } f'(x_0) = 0, \end{cases}$$

and the *Newton iteration function*,

$$N_k(f, x_0) = \underbrace{(N \circ N \circ \cdots \circ N)}_{k \text{ times}}(f, x_0).$$

That is, $N_k(f, x_0)$ is the result of k repeated applications of the Newton step function to the output of each previous evaluation. In other words, given an inital guess x_0,

$$x_1 := N(f, x_0),\ x_2 := N(f, x_1),\ \ldots,\ x_k := N(f, x_{k-1}) = N_k(f, x_0).$$

The function, N_k, satisfies the property,

$$\lim_{k \to \infty} N_k(f, x_0) = x^* \quad \text{where} \quad f(x^*) = 0.$$

That is, the limit x^* is equal to some root of the function f.

1. Write the body of the Python function `newton_step(f, df, x0)` located at the Python module `homework1.exercise1.newton_step`. (i.e. in the file `homework1/exercise1.py` within the homework repo.) To account for floating point roundoff error change the condition

$$\text{if } f'(x_0) \neq 0$$

to

$$\text{if } |f'(x_0)| < 10^{-12}$$

in the definition of $N(f, x_0)$.

2. Use your `newton_step` function to write the body of the function `newton(f, df, x0)`.