Math 456 Fall 2024
Lab 1: Iteration in one dimension
Deadline: 4 October

> She started with an equation and turned it into a graph. I've got a graph—real data—and I'm trying to find the equation that would give you the graph if you used it the way she's used hers. Iterated it.
>
> —Tom Stoppard, *Arcadia*

Reminders

- Save your work **every time** you enter new material in a cell and *before* you send it to the kernel for evaluation.
- Your submission should contain responses to the **Notebook exercises**. Every cell (or sequence of cells) in your worksheet should have, at least, a preceding comment—in a dedicated text cell—that briefly describes the evaluation cell's content and what the outcome means.
- You can develop your code and write your report with a partner. Submit one notebook for the group. Include names of group members at the top of the notebook.
- Submit by email **one** notebook file named `yourlastname_lab#.nb`
- To reduce the size of the file, use the Delete all output facility under Cell on the drop-down menu.
- The Wolfram Documentation facility is accessible in the Help menu.
- *Mathematica* code appears in `typewriter font`.

Task

Write a program that will iterate a one-dimensional map for an arbitrary starting point.

Making an iterator

What you want to create is a procedure whose *input* is

1) a map $f(x)$
2) a starting point x_0
3) a number n of iterations that produces the partial orbit

$$x_0,\ x_1 = f(x_0),\ x_2 = f(x_1),\ \ldots,\ x_n = f(x_{n-1}).$$

A standard way to do this is by means of a loop. The *loop* commands `Do` or `NestList` are appropriate for the task at hand. Using `Do`, you can make a list `orb` in which the *next* element of the list is f applied to the *current* element. That is, if you call your list of iterates `orb`, the kth element is designated `orb[k]` and you can set

`orb[k+1]` = `f(orb[k])`.

The routine

```
Do[ such-and-such, {k,kStart,kStop} ]
```

says

> Implement *such-and-such* (which depends on an index k) for values of k starting at *kStart*, ending at *kStop*, and incrementing by one after each implementation.

In the case of iteration, the *such-and-such* that you want to do is *apply* f to x_0 n times. A nice outcome is to produce a routine, called `iterate[f_,x0_,n_]` where `f`, `x0`, and `n` respectivley stand for the map, initial point, and number of iterations.

Now, use `Do` or `NestList` to create such an iteration procedure. A good programming practice is to build the procedure for a specific and simple map and, after it's working, modify it to deal with maps in general. The output of your procedure should be a list of values that give the first n points in the orbit of x_0. To do this efficiently, you should *add* the next iterate x_{n+1} to the list

$$\{x_0, x_1, \ldots, x_n\}$$

that you already have. You don't want to compute *every* value in the trajectory for each iteration. One way to achieve this result is to place the orbit elements `orb[k]` into a list using

```
Table[ orb[k], {k,kStart,kStop} ].
```

To speed up calculations, use decimal expressions for numbers. The command `N[number]` converts an exact expression into a decimal. For example,

`N[1/4]`=.25   `N[Sqrt[2]]` = 1.41....

Notebook exercises

Iterate the following maps. Compute some trajectories and see what happens. Conduct experiments to explore the following issues. Plot the graph of each map. Are there subsets of **R** that map into themselves? How can you tell? By experimenting with a variety of starting points, try to find attracting points. Are there trajectories that *aren't* attracted somewhere? One thing to check are the fixed points. You can use *Mathematica* to compute the fixed points. (Check out `Solve` and `NSolve`.) Is a given fixed point attracting? Repelling? Neither? Use both theory and experiment to describe the global dynamics. Make comparisons between maps where appropriate.

1) $x \longrightarrow x^2 - 1$

2) $x \longrightarrow \frac{1}{2}\left(x + \frac{1}{x}\right)$

3) $x \longrightarrow x^2 + x$

4) $x \longrightarrow \begin{cases} x\,\sin\left(\frac{1}{x}\right) & x \in [-1,1] - \{0\} \\ 0 & x = 0 \end{cases}$

5) $x \longrightarrow f(x) =$ your choice (make it interesting, but simple enough to analyze).