Homework 3: Nonlinear Equations: Newton, Steffensen, and Others (due on February 14)

---

1. Prove that the sequence

$$c_0 = 3\,; \quad c_{n+1} = c_n - \tan c_n\,, \quad n = 1, 2, \ldots \tag{1}$$

converges. Find the convergence limit and the order of convergence.

2. Prove that if $g(x) \in C^m$ for some $m > 1$ (continuous together with its derivatives to the order m), $g(c) = c$, $g'(c) = g''(c) = \ldots = g^{(m-1)}(c) = 0$, $g^{(m)}(c) \neq 0$, and the fixed-point iteration

$$c_{n+1} = g(c_n) \tag{2}$$

converges to c, then the order of convergence is m.

Hint: Use the Taylor series of $g(x)$ around $x = c$.

3. Determine the order of convergence for the following methods:

   (a) The *modified* Newton's method

$$c_{n+1} = c_n - m\,\frac{f(c_n)}{f'(c_n)} \tag{3}$$

   under the conditions $f(x) \in C^{m+1}$ ($m \geq 1$), $f(c) = f'(c) = f''(c) = \ldots = f^{(m-1)}(c) = 0$, and $f^{(m)}(c) \neq 0$.

   (b) Olver's method

$$c_{n+1} = c_n - \frac{f(c_n)}{f'(c_n)} - \frac{1}{2}\,\frac{f''(c_n)\,f(c_n)^2}{\left[f'(c_n)\right]^3} \tag{4}$$

   under the conditions $f(x) \in C^4$, $f(c) = 0$, and $f'(c) \neq 0$.

   (c) Steffensen's method

$$c_{n+1} = c_n - \frac{f(c_n)^2}{f\left[c_n + f(c_n)\right] - f(c_n)} \tag{5}$$

   under the conditions $f(x) \in C^2$, $f(c) = 0$, and $f'(c) \neq 0$.

4. (Programming) In this assignment, you will study the convergence of different methods experimentally using graphical tools. Note that the convergence limit

$$\lim_{n\to\infty} \frac{|c - c_{n+1}|}{|c - c_n|^p} = z \tag{6}$$

corresponds to the linear function

$$y = \log z + p\,x \tag{7}$$

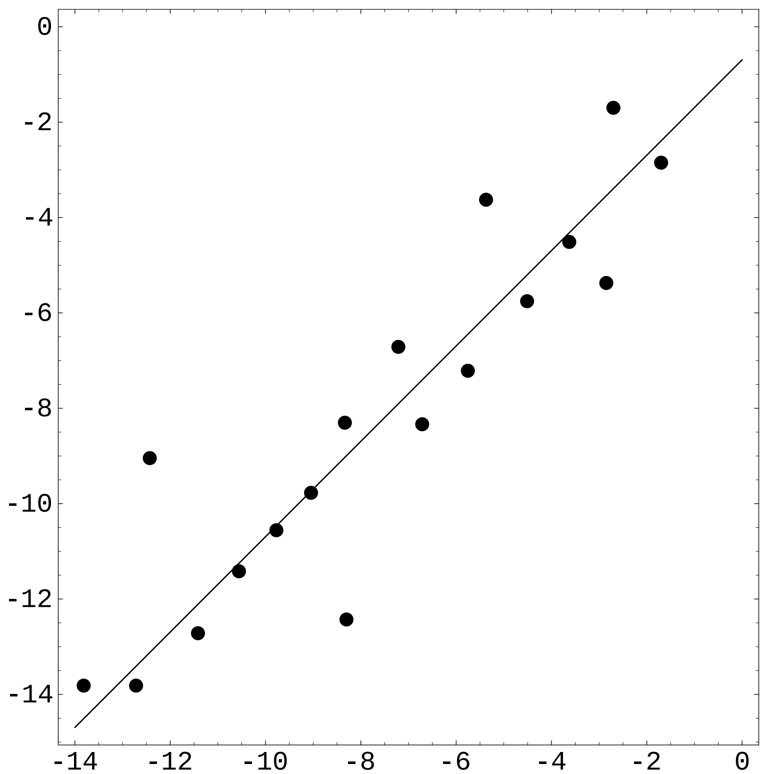
in logarithmic coordinates $x_n = \log|c - c_n|$, $y_n = |c - c_{n+1}|$. Plotting the points $\{x_n, y_n\}$ against the theoretical line verifies experimentally the order of convergence.

In the previous homework, we found that the equation

$$x + e^x = 0 \tag{8}$$

has the root at $c \approx -0.567143$ (accurate to six significant digits).

The figure shows the logarithmic plot of bisection iterations $\{x_n, y_n\}$ plotted against the line $y = \log(1/2) + x$. We can see that the iterations oscillate chaotically around the line. You will investigate whether the convergence behavior of other methods is more predictable.



Implement and apply the following methods:

(a) Fixed-point iteration. Apply it to $g(x) = -e^x$ starting with $c_0 = -1$.

(b) Newton's method. Apply it to $f(x) = x + e^x$ starting with $c_0 = -1$.

(c) Secant method. Apply it to $f(x) = x + e^x$ starting with $c_0 = 0$ and $c_1 = -1$.

In each case, find the root with the accuracy of six significant digits and plot the points x_n, y_n and the theoretical convergence line. Since some methods converge faster than others, you will need to use different number of points. Use at least 19 points for (a), 2 points for (b), and 3 points for (c).

5. (Programming) In this assignment, you will compute the motion of a planet according to Kepler's equation — one of the most famous nonlinear equations in the history of science. Kepler's equation has the form

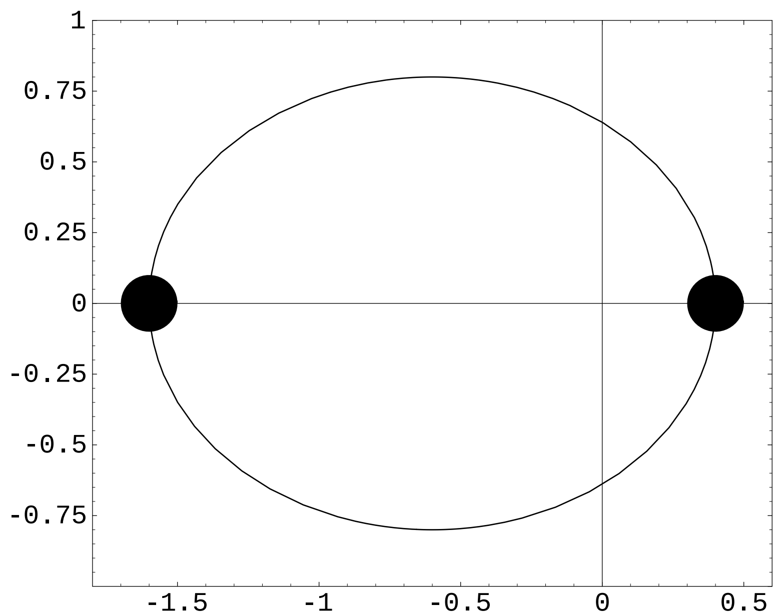
$$\omega t = \psi - \epsilon \sin\psi \, , \tag{9}$$

where t is time, ω is angular frequency, ϵ is the orbit eccentricity, and ψ is the angle coordinate. To find the planet location at time t, we need to solve equation (9) for ψ. The planet coordinates x and y are then given by

$$x = a\,(\cos\psi - \epsilon) \, ; \tag{10}$$

$$y = a\sqrt{1-\epsilon^2}\,\sin\psi \, , \tag{11}$$

where a is the major semi-axis of the elliptical orbit. For our planet, we will take $a = 1\,\mathrm{AU}$ (astronomical unit), and the eccentricity $\epsilon = 0.6$ (which is much larger than the orbit eccentricity of the Earth and other big planets in the Solar system). The picture shows the orbit and the planet positions in January ($\psi = \pi$) and July ($\psi = 0$).



Your task is to find the planet location in the other ten months, assuming that each month takes 1/12 of the rotation period. Solve Kepler's equation (9) for $\omega t = 0, \pi/6, 2\cdot\pi/6, \ldots, 11\cdot\pi/6$. You can use any numerical method to do that (either your own program or a library program). The result should be computed with the precision of 1 second (1/3600 of 1°). Output a table of the form

ωt	ψ	x	y

and then use a graphics program to plot the planet locations.