1. Consider the function

$$f(x) = 2\cos(x) + \sin(2x)\,. \tag{1}$$

Use Taylor expansions to derive three different quadratic approximations to this function. Think about using important points of the function (i.e. intercepts or stationary points) as starting points for Taylor expansion. How does the approximation differ based on the point chosen around which to Taylor expand? Feel free to use an online graphing calculator to help.

2. The time propagator for an eigenvector of the Hamiltonian operator is given by

$$|\psi(t)\rangle = c_1 e^{-iE_1 t/\hbar}\,|\phi_1\rangle + c_0 e^{-iE_0 t/\hbar}|\phi_0\rangle \tag{2}$$

where E_1, c_1, E_0, and c_0 are constants. Use Taylor series centered around $t = 0$ to find a first order and third order approximation for $|\psi(t)\rangle$. When might the first order approximation be appropriate? When might you want to use the higher order approximation?

3. Suppose the derivative of a function is given by

$$\frac{df}{dt} = f(t) - t \tag{3}$$

with the initial condition $f(t = 0) = \frac{1}{2}$.

Write down a pseudo-code (a list of steps in plain language) for approximating the function on the interval 0 to 1. What parameter(s) will you need to specify to make the approximation, and how will they affect the accuracy? Can you think of multiple ways to perform the approximation? Are there trade-offs between accuracy and speed?

Keep a copy of the pseudo-code because on Python day you will turn it into real code.

4. Solve the self-consistent equation

$$x = \tanh(ax)\,, \tag{4}$$

where a is a parameter that is greater than 0. How do the solutions (and in particular the number of real solutions) change with different values of a? Again, a graphing calculator might come in handy.