Lab Four: 5 Jan. 2012
Computational Physics

Lab Assignment. Finite square well. These exercises are due on 13 January 2012. You may work in groups of two or three if you want. There should not be more than three people on a lab report. Be sure to complete all the matlab calculations in lab today. There will not be time in lab to work on this next week.

For all of these problems, keep the width fixed at $a = 0.3$ nm. For each question, include a short explanation of how you solved the problem. There is no need to include matlab code, but be sure you understand what it is that matlab

1. For the infinite square well, calculate by hand the lowest two energy levels. The lowest energy level is associated with an even wave function, and the next lowest energy is associated with an odd wave function.
2. Use the `even` and `odd` functions that you worked with last time. Set $V_0 = 1000000$. This corresponds to an approximately infinite well. Make plots of `even(E)` and `odd(E)` and find the approximate location of the lowest solution. Then use the `fzero` command to find a more exact answer for the lowest energy level. Compare this with your answers to the previous questions. Is your answer what you would expect? Explain briefly.
3. Determine all allowed energies if $V_0 = 20$. (Remember that our equations do not hold if $E > V_0$.) Find the allowed energies for both even and odd wave functions.
4. (Optional/bonus problem.) For what approximate range of V_0 are there exactly three bound states?

When you are done with these calculations you may go if you want. But if you have energy, I could suggest starting the following set of exercises, which we will do next week in lab.

Euler's Method. I have prepared a short matlab function that will calculate and make plots of approximate solutions to differential equations obtained via Euler's method. To see how to use this code, do the following:

1. Find the file `euler.m` from `http://hornacek.coa.edu/dave/KIST`. Copy this code into a file on your computer. The file must have the name `euler.m`.
2. Read over the matlab code you just downloaded. The comments in the code should explain what the function does and how to use it.
3. The `euler` function finds approximate solutions to differential equations of the following form:
$$y'(t) = f(y,t) . \tag{1}$$
You will need to define the function $f(y,t)$. Do so with the `inline` command. For example, to solve the equation we discussed in class and that is in the notes from class four, type
```
f = inline('-t*y', 't', 'y')
```
4. You also need to tell matlab the initial and final values for t, the initial condition for $y(t)$, and the number of steps. Doing so then determines the step size Δt. If the initial t value is a and the final value is b, then
$$\Delta t = \frac{b-a}{n} . \tag{2}$$
For the example from class, $b = 2$, $a = 0$, $y(0) = 2$ and $n = 8$. To get the Euler solution, type the following:
```
euler(f, [0,2], 2, 8)
```
The general command is
```
euler(f, [initial_t,final_t], initial_condition, number_of_steps )
```
5. Modify the above command so that $\Delta t = 0.1$.

Now let's use the `euler` function to solve another differential equation:

$$y\prime(t) = \cos(t)\sin(y) , \quad y(0) = 15 . \tag{3}$$

1. Have matlab solve this equation using Euler's method from $t = 0$ to $t = 20$. Start with a small n value.
2. Determine a good n value as follows. Increase n, observing the plot of the solution. Keep increasing n until the plot does not change significantly.

Next week you will complete and hand in some exercises where you use `euler` to solve some differential equations that arise from a physics problem.