PHYS 391, Homework #3
Due Tuesday, November 3, start of class.

Key Concepts

- Outlier Rejection
- Weighted Mean
- Least Squares Fitting

Reading: Taylor Chapters 6-8

Homework Problems (taken from Taylor):

1. Problem 6.7 - This is a good cautionary tale against rejection of data in small samples.
2. Problem 7.1 - Please work this out "by hand." There aren't that many values, and it is good to see exactly how this works at least once.
3. Problem 7.6
4. Problem 7.7
5. Problem 8.1 - Do this "by hand" first using the appropriate equations, then compare with the python function `numpy.polyfit(x, y, 1)` which does a least-squares fit to a 1st order polynomial. See the `numpy.polyfit` documentation for more details on the arguments to this function. You don't need 'squared paper' but do please draw a neat graph.
6. Problem 8.10 - In addition to the values A and B, you should also calculate the uncertainties on A and B (see problem 8.19 for the relevant equations). Here you can do the weighted fit any way you want. Either by hand, by performing the calculations in python, or perhaps most useful try to learn the built-in python functions to do this. To fit weighted data, you need to supply more arguments to the `numpy.polyfit` function. Note that `polyfit` wants a vector of *weights* not errors. The errors on the fitted parameters can be returned as a *covariance matrix* (see Chap. 9), and if we don't get to this next week, come talk to me. If you do use python, be sure to check your results with another method!
7. Problem 8.20 - Review the arguments in section 8.5 to see how the errors can be "bootstrapped" from the data itself. Note that one assumption of this technique is that the errors on each point are equal.
8. Problem 8.25 - Here you do not need to worry about the errors (that would be Problem 8.26 and there are some subtleties in this approach we will cover in Lab 4). Be careful in how you linearize this, but fundamentally you can do a non-weighted linear fit.